endace
accelerated

DAG 7.1S Card User Guide

EDM01-17

www.endace.com

Published by:

Endace Limited

Building 7

Lambie Drive

PO Box 76802
Manukau City 1702
New Zealand

Phone: +64 9 262 7260

Fax: +64 9 262 7261

support@endace.com

www.endace.com

International Locations

New Zealand
Endace Technology® Ltd

Level 9
85 Alexandra Street

PO Box 19246
Hamilton 2001
New Zealand

Phone: +64 7 839 0540
Fax: +64 7 839 0543

Americas
Endace USA® Ltd

Suite 220
11495 Sunset Hill Road

Reston
Virginia 20190
United States of America

Phone: ++1 703 382 0155
Fax: ++1 703 382 0155

Europe, Middle East & Africa
Endace Europe® Ltd

Sheraton House
Castle Park

Cambridge CB3 0AX
United Kingdom

Phone: ++44 1223 370 176
Fax: ++44 1223 370 040

Protection Against Harmful Interference

When present on equipment this manual pertains to, the statement "This device complies with part 15 of the FCC rules" specifies the equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the Federal Communications Commission [FCC] Rules.

These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment.

This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications.

Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

Extra Components and Materials

The product that this manual pertains to may include extra components and materials that are not essential to its basic operation, but are necessary to ensure compliance to the product standards required by the United States Federal Communications Commission, and the European EMC Directive. Modification or removal of these components and/or materials, is liable to cause non compliance to these standards, and in doing so invalidate the user’s right to operate this equipment in a Class A industrial environment.

Disclaimer

Whilst every effort has been made to ensure accuracy, neither Endace Limited nor any employee of the company, shall be liable on any ground whatsoever to any party in respect of decisions or actions they may make as a result of using this information.

Endace Limited has taken great effort to verify the accuracy of this manual, but assumes no responsibility for any technical inaccuracies or typographical errors.

In accordance with the Endace Limited policy of continuing development, design and specifications are subject to change without notice.

Table of Contents

Chapter 1: Introduction

Overview

The Endace DAG 7.1S card provides the means to transfer data at the full speed of the network into the memory of the host PC, with zero packet loss guaranteed in even worst-case conditions. Further, unlike a NIC, Endace products actively manage the movement of network data into memory without consuming any of the host PC's resources. The full attention of the CPU remains focused on the analysis of incoming data without a constant stream of interruptions as new packets arrive from the network. For a busy network link, this feature has a turbo-charging effect similar to that of adding a second CPU to the system.

The DAG 7.1S is a network monitoring interface card specifically designed to provide high efficiency monitoring and transmission of ATM, POS or Bit HDLC traffic with precision timestamping capability.

It supports the following:

- Concatenated POS/ATM receive and transmit over 4 x OC-3c/STM-1c or 4 x OC-12c/STM-4c.
- Channelised Bit HDLC/ATM receive and transmit over 4 x OC-3/STM-1 or 2 x OC-12/STM-4.

Purpose of this User Guide

Description

The purpose of this User Guide is to provide you with an understanding of the DAG card architecture and functionality and to guide you through the following:

- Installing the card and associated software and firmware,
- Configuring the card for your specific network requirements,
- Running a data capture session,
- Synchronising clock time,
- Data formats

You can also find additional information relating to functions and features of the DAG 7.1S card in the following documents which are available from the Support section of the Endace website at www.endace.com:

- EDM04-08 Configuration and Status API Programming Guide
- EDM04-13 SAR API Programming Guide
- EDM04-11 IXP Filter API Programming Guide
- EDM04-08 DAG IXP Filter Loader User Guide

This User Guide and the Linux and Window Guides are also available in PDF format on the Installation CD shipped with your DAG 7.1S card.

System Requirements

General

The minimum system requirements for the DAG 7.1S card are :

- PC, at least Intel Xeon 1.8GHz or faster
- Minimum of 256 MB RAM
- At least one free PCI-Express slot supporting at least one lane
- Software distribution requires 30MB free space
- 6GB for installation of Endace software, which is optional

Operating System

This User Guide assumes you are installing the DAG card in a PC which already has an operating system installed.

However for convenience, a copy of Debian Linux 3.1 (Sarge) is provided as a bootable ISO image on the CDs that is shipped with the DAG card.

To install either the Linux/FreeBSD or Windows operating system please refer to the following documents which are also included on the CD that is shipped with the DAG card.

- EDM04-01 Linux FreeBSD Software Installation Guide
- EDM 04-02 Windows Software Installation Guide

Other Systems

For advice on using an operating system that is substantially different from either of those specified above, please contact Endace Customer Support at support@endace.com

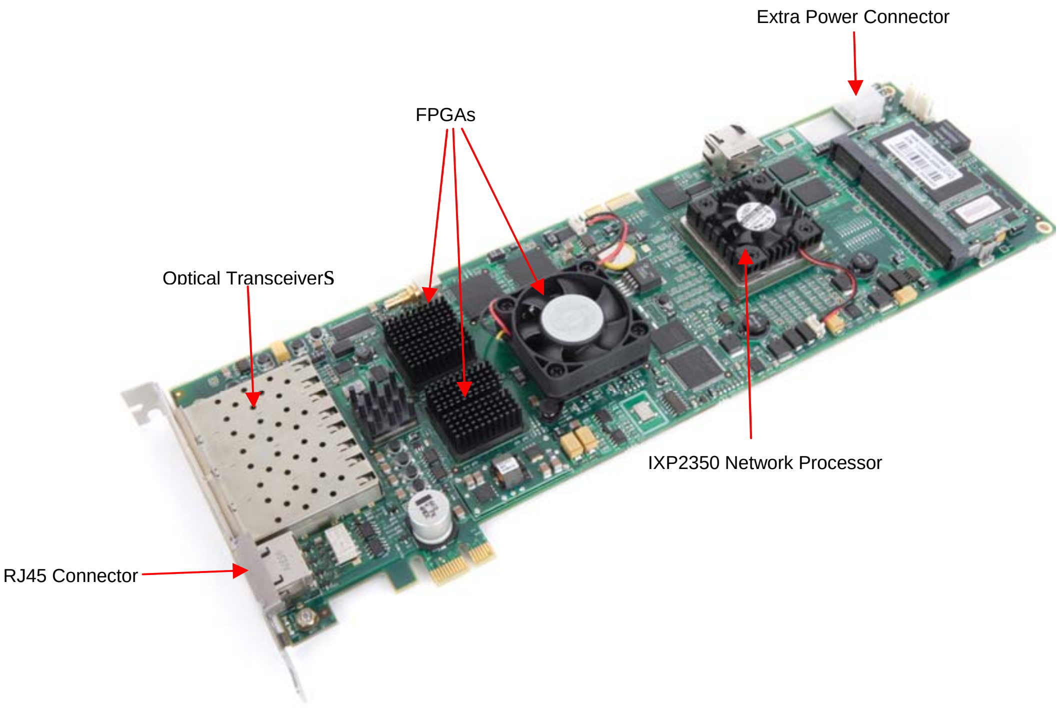
Card Description

The DAG 7.1S SDH/SONET Network Monitoring Card provides either four STM-1 (OC3) or two STM-2 (OC12) interfaces supporting concatenated or channelised ATM or Packet Over Sonet (POS) networks.

The DAG 7.1S has four optical transceivers which can be operated simultaneously.

The key features of the card are:

- Four interfaces allow full line rate capture and processing for 4 x STM-1/OC-3 or 2 x STM-2/OC-12.
- Fully programmable Intel IXP Network Processor
- PCI Express bus interface.
- 1244Mpps raw transmit and receive bandwidth.
- Combined FPGA and network processor architecture.
- Channelised and concatenated support.
- ATM AAL2 and AAL5 segmentation and reassembly.
- PoS IP filtering



Card Architecture

Serial SONET/SDH optical data is received by four optical interfaces, and passed through deserializers.

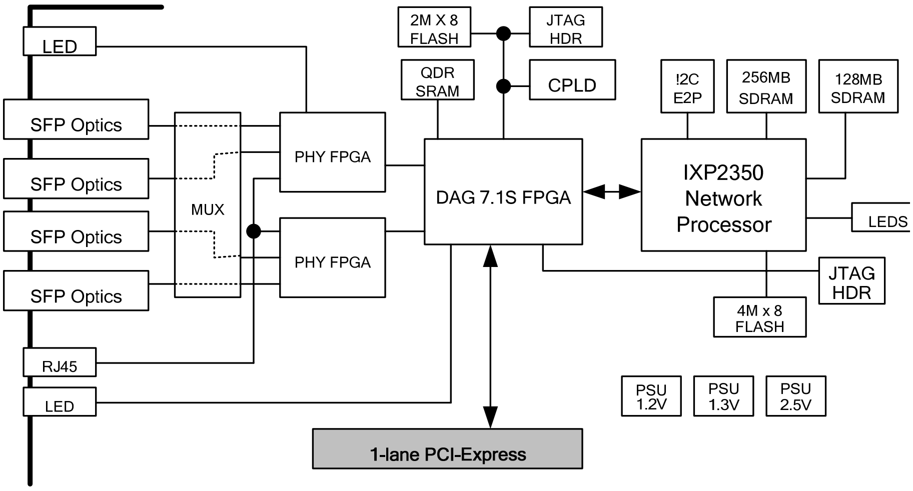
The network data feeds immediately into two physical layer FPGAs. The SONET/SDH payload data is then sent to the main FPGA.

The FPGA contains the packet record processor, PCI Express interface logic and the DAG Universal Clock Kit (DUCK) timestamp engine. The DUCK provides high resolution per packet timestamps which can be accurately synchronised..

> **Note:** For further information on the DUCK and time synchronising please refer to *Chapter 7: Synchronising Clock Time* later in this User Guide.

An Intel IXP network processor is logically located next to the main FPGA. The main FPGA can route packets to either the IXP network processor for additional processing before routing onto the host or directly to the host via the PCI-Express port.

The following diagram shows the card's major components and the flow of data.



Extended Functions

In addition to standard packet capture the DAG 7.1S card also provides TCP/IP Filtering and Classification and ATM Segmentation and Reassembly

TCP/IP Filtering and Classification

This feature allows you to classify packets into arbitrary categories which then drop, retransmit or capture a packet to the host based upon the result. You can also change filter rules "on the fly" with any loss of data

The specifications for the IP filtering/packet classification are:

- Packets are classified and filtered by IP header (both IPv4 and IPv6) and/or UDP/TCP/SCTP port number.
- Up to 1024 IP header classification rules.
- Up 254 UDP/TCP/SCTP port or ICMP type rules can set per IP header classification.
- Classification rules are assigned a user-defined 14-bit identifier
- Packets matching classification rules are assigned the matching rule's identifier.
- Programmable actions may be associated with each rule identifier. For example the packet should either be dropped, or presented to the host.
- Packets presented to the host include the rule-match identifier in the record header.

AAL2/AAL5 Reassembly

This feature allows you to eliminate the significant CPU load associated with AAL2/AAL5 reassembly on a busy ATM link by offloading this process to the DAG card. It also provides the ability to reduce volume of captured data to only what is required by filtering on VPI/VCI pairs.

The Reassembler specifications are:

- Supports up to 8160 simultaneously active VCI/VPI/CIDs
- Supports simultaneous reassembly of AAL2 and AAL5 frames up to 8kB long.
- VPI/VCI scanning
- Supports up to full STM-4/OC-12 cell rate on two interfaces simultaneously (approx 2.8 million cells/sec), or four full STM-1/OC-3 interfaces for AAL 5 reassembly.
- Supports 2 x STM-1/OC-3 cell rate on combined four interfaces (approx 0.8 million cells/sec) for AAL2 reassembly.
- Optional ATM cell filtering prior to reassembly.

Chapter 2: Installation

Introduction

A DAG 7.1S card can be installed in any free PCI-Express slot.

The DAG 7.1S card operates on a single lane PCI-Express, this interface is capable of providing a maximum throughput of 1.8Gbits/s for both receive and transmit.

You can run multiple DAG 7.1S cards on one bus. By default, the DAG driver supports up to four DAG cards in one system.

DAG Driver Device

The DAG device driver must be installed before you install the DAG card itself.

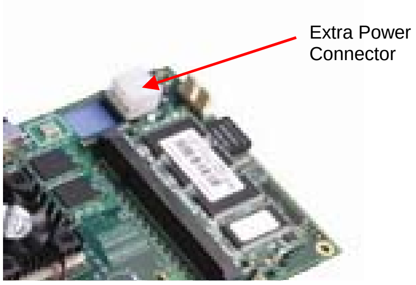
If you have not already completed this please follow the instructions in *EDM04-01 Linux FreeBSD Software Installation Guide* or E*DM 04-02 Windows Software Installation Guide* as appropriate, which are included on the CD shipped with the DAG card.

Inserting the DAG Card

To insert the DAG card in the PC follow the steps described below:

- Turn power to the computer OFF,
- Remove the PCI bus slot screw and cover,
- Insert DAG card into PCI-e bus slot ensuring that it is firmly seated in the slot,
- Check the free end of the card fits securely into the card-end bracket that supports the weight of the card,
- Secure the card with the bus slot screw,
- Connect the extra power connector located on the top edge of the card.

! **Note:** Ensure you do this before powering up the computer. Failure to do so may cause damage to the card.



- Turn power to the computer ON.

Port Connectors

The DAG 7.1S has 4 SFP socket connectors. Each connector consists of an optical fibre transmitter and receiver.

The upper connector of each pair is used for the transmit signal. These can be connected to daisy-chain systems if you have facility loopback (fcl) set on the card. You can also connect them if you are using a data generation programme.

The bottom connector of each pair is used for the received signal.

There is an 8-pin RJ-45 socket located below the optical port connectors on the car bracket. This is available for connection to an external time synchronisation source.

Caution: Never connect an Ethernet network or telephone line to the RJ-45 sockets.

Pluggable Optical Transceivers

Overview

The DAG7.1S card uses industry standard Small Form-factor Pluggable (SFP) optical transceivers.

The transceivers consists of two parts:

- Mechanical chassis attached to the circuit board
- Transceiver unit which may be inserted into the chassis

Note: You must select the correct transceiver type to match the optical parameters of the network to which you want t connect. Configuring the card with the wrong transceiver type may damage the card.

You can connect the transceiver to the network via LC-style optical connectors.

For further information on Pluggable Optical Transceiver please refer to the Endace website at www.endace.com/dagpluggable.htm.

Setting Power

The optical power range depends on the particular SFP module that is fitted to the DAG card.

However Endace recommends the SFP modules described below which can be supplied with the DAG 7.1S card:

Manufacturer	Part number
Finisar	FTRJ1322
Finisar	FTRJ1323
Optical Communication Products	TRPD12MM3EAS
Optical Communication Products	TRPD03MM3EAS

Pluggable Optical Transceivers (cont.)

Optical power is measured in dBm. This is decibels relative to 1 mW where 10 dB is equivalent to a factor of 10 in power.

The optical power is always a negative value, indicating power that is less than 1 mW. The most sensitive devices can work at power levels down as low as –30dBm or 1µW.

The DAG 7.1S card optical power module configuration for Multi Mode Fibre (MMF) and Single Mode Fibre (SMF) is shown below:

Part #	Fibre	Data Rate	Max Pwr	Min Pwr	Nom Pwr	Mode
FTRJ1322	SMF	622	-8dBm	-28dBm	-	OC-12 Single Mode
FTRJ1323	SMF	155	-8dBm	-28dBm	-	OC-3 Single Mode
TRPD12MM3EAS	MMF	622	-	-	-	OC-12 Multi Mode
TRPD03MM3EAS	MMF	155	-	-	-	OC-3 Multi Mode

Power Input

Note: The optical power input to the DAG card must be within the receiver’s dynamic range of 0 to -22dBm. If it is slightly outside of this range it will cause an increased bit error rate. If it is significantly outside of this range the system will not be able to lock onto the SONET signal.

When power is above the upper limit the optical receiver saturates and fails to function. When power is below the lower limit the bit error rate increases until the device is unable to obtain lock and fails. In extreme cases, excess power can damage the receiver.

When you set up the DAG card you should measure the optical power at the receiver and ensure that it is within the specified power range. If it is not, adjust the input power as follows:

- Insert an optical attenuator if power is too high, or
- Change the splitter ratio if power is too high or too low.

Splitter Losses

Splitters have the insertion losses either marked on their packaging or described in their accompanying documentation. General guidelines are:

- A 50:50 splitter will have an insertion loss of between 3 dB and 4 dB on each output
- 90:10 splitter will have losses of about 10 dB in the high loss output, and <2 dB in the low loss output

Note: A single mode fibre connected to a multi-mode input will have minimal extra loss. However a multi-mode fibre connected to a single mode input will create large and unpredictable loss.

Chapter 3: Configuring the Card

Introduction

Configuring the DAG card for data capture involves the following steps:

- Loading the images and programming the FPGAs,
- Setting the link,
- Checking the link,
- Configuring the connections,
- Capturing data.

The DAG 7.1S card uses four integrated SONET/SDH ATM/PoS physical layer interface devices to support capture of ATM cells or Bit HDLC and PoS data frames.

The `dagconfig` tool which is also supplied with the DAG card allows you to configure the card to your specific network requirements as well as view interface statistics. Sample `dagconfig` outputs are shown later in this chapter.

Line Characteristics

Overview

It is important that you understand the physical characteristics of the network to which you want to connect before you begin configuring the card.

Because of its flexibility the card will accept a wide range of settings. However if they are not the correct settings for your network, the card will not function as expected.

Note: If you are unsure about which of the options listed below apply to your network, please contact your Network Administrator for further information before proceeding with configuring the card.

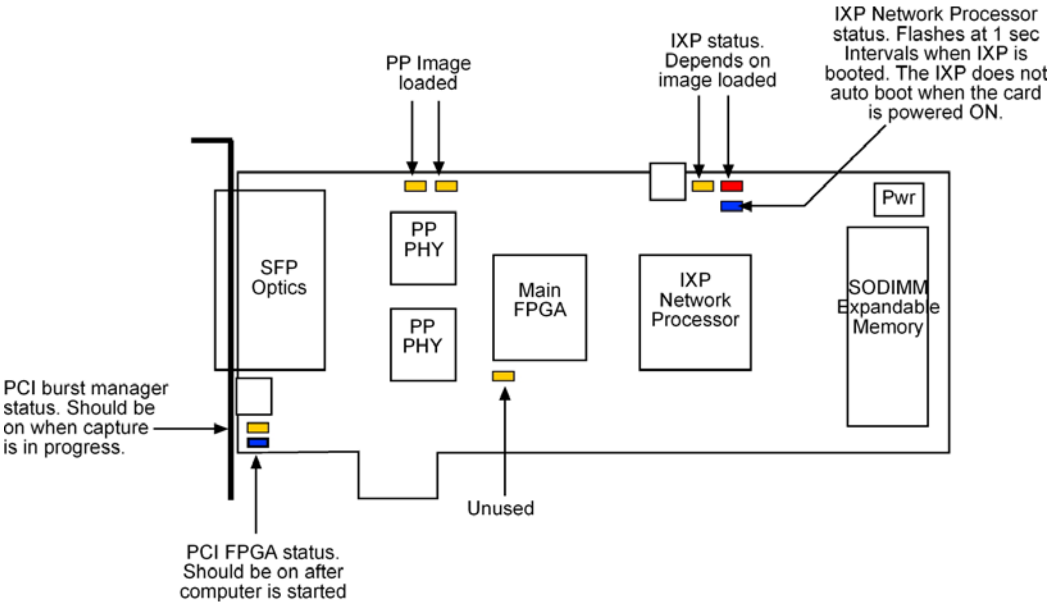
Supported Line Types

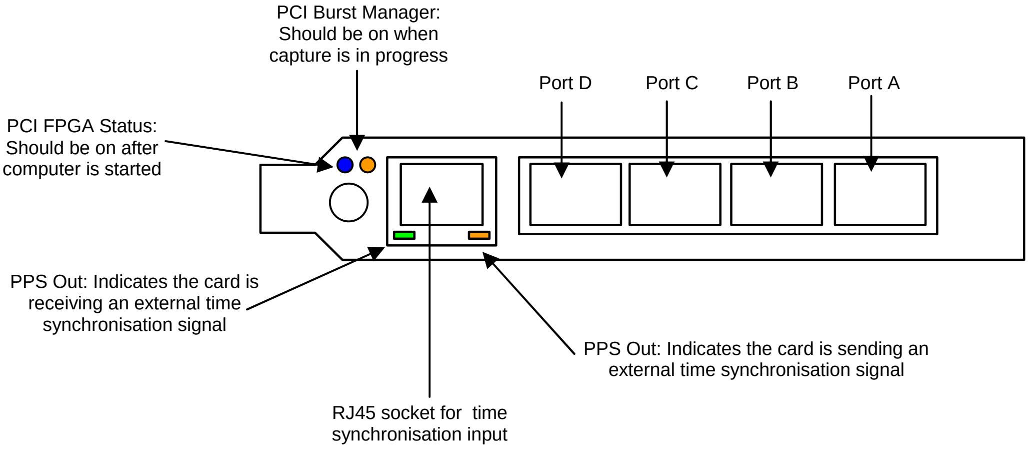
The line characteristics supported by the DAG 7.1S card are described below.

PoS/ATM	Packet over SONET/Asynchronous Transfer Mode.
OC-3/OC-3c	A SONET network line with transmission speeds up to 155.52 Mbit/s using fiber optics. Also called STM-1 (SDH).
OC-12/OC-12c	A SONET network line with transmission speeds up to 622.08 Mbit/s using fiber optics. Also called STM-4 (SDH).
STM-1/STM-1c	An SDH line equivalent to OC-3 (SONET).
STM-4/STM-4c	An SDH line equivalent to OC-12 (SONET).

LEDs and Inputs

Before you begin to configure the DAG card it is important to understand the function of the various LEDs associated with the card, as well as the sockets on the PCI bracket.


PP Image loaded
IXP status. Depends on image loaded
IXP Network Processor status. Flashes at 1 sec Intervals when IXP is booted. The IXP does not auto boot when the card is powered ON.
Pwr
SFP Optics
PP PHY
PP PHY
Main FPGA
IXP Network Processor
SODIMM Expandable Memory
PCI burst manager status. Should be on when capture is in progress.
Unused
PCI FPGA status. Should be on after computer is started


PCI Burst Manager: Should be on when capture is in progress
Port D
Port C
Port B
Port A
PCI FPGA Status: Should be on after computer is started
PPS Out: Indicates the card is receiving an external time synchronisation signal
PPS Out: Indicates the card is sending an external time synchronisation signal
RJ45 socket for time synchronisation input

Receiver Port Signal Levels

! **Caution:** Ensure that you insert only OC-3/OC-12 modules. Do not use OC-48 or GiG-E modules as these will destroy the PHY FPGA

The card supports 1310 nanometer singlemode and multimode fibre attachments with optical signal strength between 0 dBm and -22 dBm.

If there is doubt, check card receiver ports light levels are correct using an optical power meter.

The card receiver ports are the lower of each dual-LC-style connectors, the closest to the PCI-Express slot.

Cover card transmit ports with LC-style plugs to prevent dust and mechanical hazards damaging optics if not in use.

Note: If you remove the optics modules for any reason they will not automatically power on when re-inserted. You will need to turn them on using the `sfppwr` option when you configure the card channelised or concatenated operation.

Specific Network Configuration

For detailed information on configuring the DAG 7.1S for data capture on concatenated and channelised networks please refer to *Chapter 4: Concatenated Configuration* and *Chapter 5: Channelised Configuration*.

Chapter 4:
Concatenated Configuration

Load the FPGA Images

Note: Before you can configure the card for capture you must first load the card with the appropriate FPGA images for the type of data you want to capture. It is important you understand the protocol used by the network to which you want to connect. If you do not load the correct image for the protocol you will not be able to capture data.

- Place the card in "`eql`" mode to prevent any erroneous signals interfering with the images during loading using :

  `dagconf –d0 eql` (where "`0`" is the device number of the DAG card)
- Load the concatenated PCI-Express FPGA image:

  ```
  dagrom -rvp –d0 -f xilinx/dag71spci-conc-terf.bit
  ```
- Load the concatenated PHY FPGA image:

  ```
  dagld –x –d0 xilinx/dag71spp-conc-terf.bit:
  xilinx/dag71spp-conc-terf.bit
  ```

Available Configurations

The available concatenated configurations are shown below:

Number of Ports	Line Type	VC Type and Number	Protocol
4	OC-3c/STM-1c	4 x VC-4	PoS/ATM
4	OC-12c/STM-4c	4 x VC-4-4c	PoS/ATM

Display Current Configuration

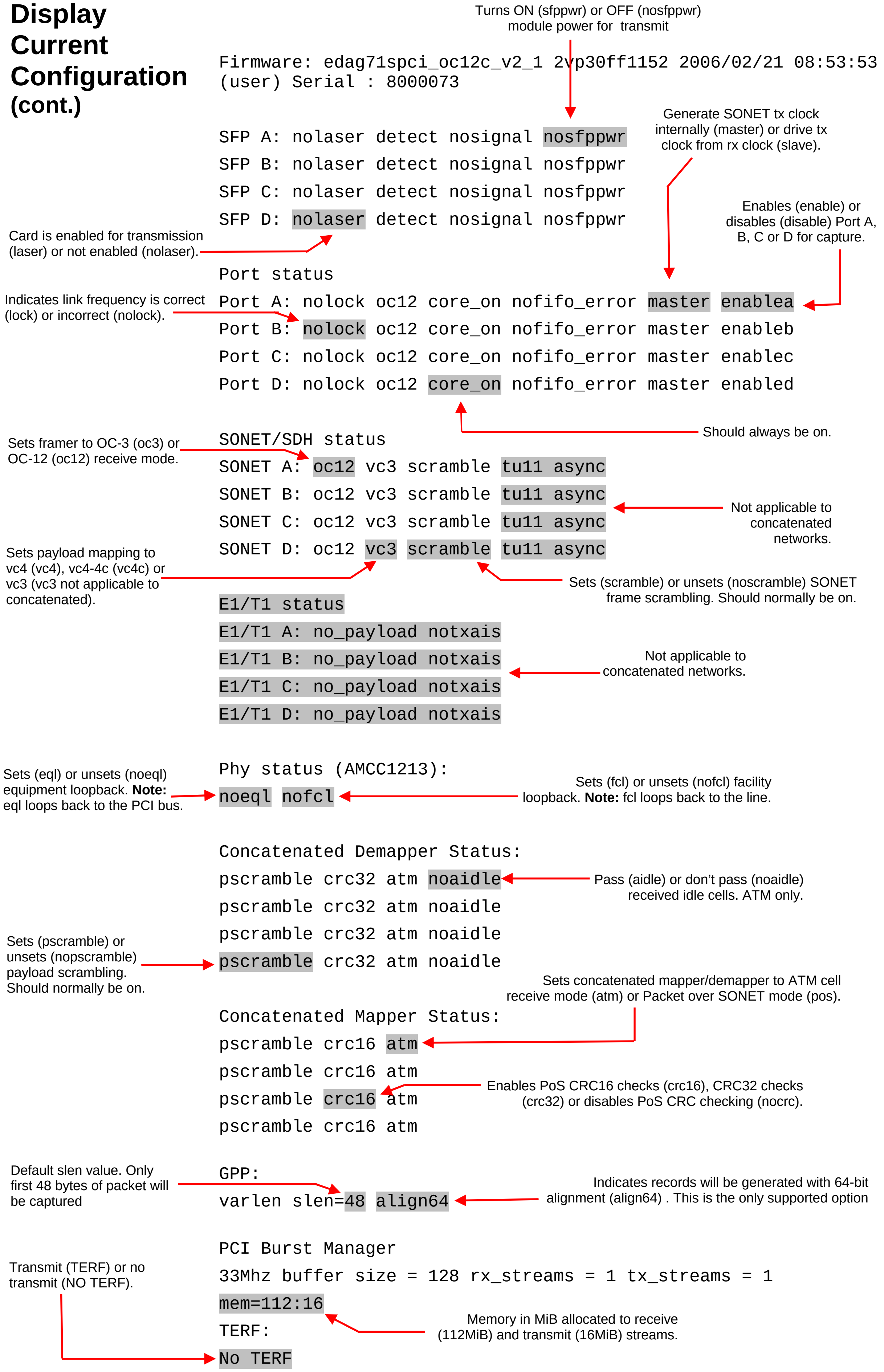
Once you have loaded the appropriate images you should run the `dagconfig` tool without arguments to display the current card configuration and verify the firmware has loaded correctly, using:

`dagconfig –d0` (where "`0`" is the device number of the DAG card)

An explanation of the default `dagconfig` output is shown on the following page:

Note: The `dagconfig` default output may include some information that is not applicable to concatenated networks and these are indicated in the example

Display Current Configuration (cont.)



```
Firmware: edag71spci_oc12c_v2_1 2vp30ff1152 2006/02/21 08:53:53
(user) Serial : 8000073

SFP A: nolaser detect nosignal nosfppwr
SFP B: nolaser detect nosignal nosfppwr
SFP C: nolaser detect nosignal nosfppwr
SFP D: nolaser detect nosignal nosfppwr

Port status
Port A: nolock oc12 core_on nofifo_error master enablea
Port B: nolock oc12 core_on nofifo_error master enableb
Port C: nolock oc12 core_on nofifo_error master enablec
Port D: nolock oc12 core_on nofifo_error master enabled

SONET/SDH status
SONET A: oc12 vc3 scramble tu11 async
SONET B: oc12 vc3 scramble tu11 async
SONET C: oc12 vc3 scramble tu11 async
SONET D: oc12 vc3 scramble tu11 async

E1/T1 status
E1/T1 A: no_payload notxais
E1/T1 B: no_payload notxais
E1/T1 C: no_payload notxais
E1/T1 D: no_payload notxais

Phy status (AMCC1213):
noeql nofcl

Concatenated Demapper Status:
pscramble crc32 atm noaidle
pscramble crc32 atm noaidle
pscramble crc32 atm noaidle
pscramble crc32 atm noaidle

Concatenated Mapper Status:
pscramble crc16 atm
pscramble crc16 atm
pscramble crc16 atm
pscramble crc16 atm

GPP:
varlen slen=48 align64

PCI Burst Manager
33Mhz buffer size = 128 rx_streams = 1 tx_streams = 1
mem=112:16
TERF:
No TERF
```

Turns ON (sfppwr) or OFF (nosfppwr) module power for transmit

Generate SONET tx clock internally (master) or drive tx clock from rx clock (slave).

Enables (enable) or disables (disable) Port A, B, C or D for capture.

Card is enabled for transmission (laser) or not enabled (nolaser).

Indicates link frequency is correct (lock) or incorrect (nolock).

Should always be on.

Sets framer to OC-3 (oc3) or OC-12 (oc12) receive mode.

Not applicable to concatenated networks.

Sets payload mapping to vc4 (vc4), vc4-4c (vc4c) or vc3 (vc3 not applicable to concatenated).

Sets (scramble) or unsets (noscramble) SONET frame scrambling. Should normally be on.

Not applicable to concatenated networks.

Sets (eql) or unsets (noeql) equipment loopback. **Note:** eql loops back to the PCI bus.

Sets (fcl) or unsets (nofcl) facility loopback. **Note:** fcl loops back to the line.

Pass (aidle) or don't pass (noaidle) received idle cells. ATM only.

Sets (pscramble) or unsets (nopscramble) payload scrambling. Should normally be on.

Sets concatenated mapper/demapper to ATM cell receive mode (atm) or Packet over SONET mode (pos).

Enables PoS CRC16 checks (crc16), CRC32 checks (crc32) or disables PoS CRC checking (nocrc).

Default slen value. Only first 48 bytes of packet will be captured

Indicates records will be generated with 64-bit alignment (align64) . This is the only supported option

Transmit (TERF) or no transmit (NO TERF).

Memory in MiB allocated to receive (112MiB) and transmit (16MiB) streams.

Display Current Configuration (cont.)

If the firmware has not loaded correctly the `dagconfig` output will indicate "`component not found`" as the `SONET/SDH status` and `E1/T1 status` as shown below:

```
SONET/SDH status
SONET A:  component not found.
SONET B:  component not found.
SONET C:  component not found.
SONET D:  component not found.

E1/T1 status
E1/T1 A:  component not found.
E1/T1 B:  component not found.
E1/T1 C:  component not found.
E1/T1 D:  component not found.
```

In this case you should perform the following steps:

- Check the firmware image that you are using and ensure that it is correct for your network configuration and protocol,
- Then reload the firmware as described in *Load the FPGA Images* earlier in this chapter.

If after performing these steps `dagconfig` still displays "`component not found`" please contact Endace Customer Support at support@endace.com for further assistance.

Verify Optical Signal

As shown on the previous page, the `dagconfig` output displays whether or not a port is receiving a valid optical signal.

Valid signal received

```
SFP A: laser detect signal nosfppwr
SFP B: laser detect nosignal nosfppwr
```

No valid signal received

To receive a valid optical signal, both ends of the link must use be using the same type of optical transceivers. In addition the optical fiber used must match the requirements of the optical transceiver and be in good condition.

Verify Mapping/ Framing Setup

Parameters

As shown on the previous page, the `dagconfig` output displays whether or not mapping and framing is setup correctly.

For correct setup both ends of the link must match the framing and mapping parameters. The main parameters involved in setting the framing are:

- Link speed (`oc3/oc12`)
- Clock master (`master/slave`)
- Payload mapping (`vc4c/vc4`)

```
Port status
Port A: lock oc12 core_on nofifo_error slave
Port B: nolock oc12 core_on nofifo_error slave
```

Link setup correctly

Link not setup correctly

```
SONET/SDH status
SONET A: oc12 vc4c scramble tu11 async
SONET B: oc12 vc4c scramble tu11 async
SONET C: oc12 vc4c scramble tu11 async
SONET D: oc12 vc4c scramble tu11 async
```

Not applicable to concatenated networks

Different Port Configurations

The DAG card allows different ports to have different configurations. For example if you want to configure only Ports A and B for OC-3c VC-4c you could do so using the following command:

```
dagconfig –d0 default -1 -4 oc3 vc4

Port status
Port A: nolock oc3 core_on nofifo_error master enablea
Port B: nolock oc3 core_on nofifo_error master enabled
Port C: nolock oc12 core_on nofifo_error master enablec
Port D: nolock oc12 core_on nofifo_error master enablec

SONET/SDH status
SONET A: oc3 vc4 scramble tu11 async
SONET B: oc3 vc4 scramble tu11 async
SONET C: oc12 vc4 scramble tu11 async
SONET D: oc12 vc4 scramble tu11 async
```

Not applicable to concatenated networks

ATM Mode

When configuring the DAG card for ATM mode ensure you configure it to match the characteristics of the network you wish to monitor.

Please refer to *Display Current Configuration* earlier in this chapter for a full explanation of `dagconfig` outputs.

> **!** **Note:** The order in which the configuration options appear in the command line is important. Therefore you should use `default` and `reset` before other options in the command line.

The example below has been configured for OC-12c VC-4-4c on ports A and B with ports C and D unused:

```
dagconfig -d0 reset default atm oc12 vc4c sfppwr
```

Note position in command line.

VC-4-4c

Turns the optics modules on if they have previously been turned off.

```
Firmware: edag71spci_oc12c_pci_v2_23 2vp30ff1152
2006/02/21 08:53:53 (user) Serial : 8000073

SFP A: nolaser detect signal sfppwr
SFP B: nolaser detect signal sfppwr
SFP C: nolaser detect nosignal sfppwr
SFP D: nolaser detect nosignal sfppwr

Port status
Port A: lock oc12 core_on nofifo_error master enablea
Port B: lock oc12 core_on nofifo_error master enableb
Port C: nolock oc12 core_on nofifo_error master enablec
Port D: nolock oc12 core_on nofifo_error master enabled
```

Link setup correctly

Link not setup correctly

```
SONET/SDH status
SONET A: oc12 vc4c scramble tu11 async
SONET B: oc12 vc4c scramble tu11 async
SONET C: oc12 vc4c scramble tu11 async
SONET D: oc12 vc4c scramble tu11 async
```

Not applicable to concatenated networks

```
E1/T1 status
E1/T1 A: no_payload notxais
E1/T1 B: no_payload notxais
E1/T1 C: no_payload notxais
E1/T1 D: no_payload notxais
```

Not applicable to concatenated networks

```
Phy status (AMCC1213):
noeql nofcl

Concatenated Demapper Status:
pscramble crc32 atm noaidle
pscramble crc32 atm noaidle
pscramble crc32 atm noaidle
pscramble crc32 atm noaidle
```

Not applicable to ATM

ATM Mode (cont.)

```
Concatenated Mapper Status:
pscramble crc16 atm
pscramble crc16 atm
pscramble crc16 atm
pscramble crc16 atm
```

Not applicable to ATM

```
GPP:
varlen slen=48 align64

PCI Burst Manager
33Mhz buffer size = 128 rx_streams = 1 tx_streams = 1
mem=112:16
TERF:
No TERF
```

PoS Mode

To configure the DAG card for PoS mode you must replace `oc12` and `vc4c` to match your network configuration as shown below:

Note: The order in which the configuration options appear in the command line is important. Therefore you should use `default` and `reset` before other options in the command line.

Please refer to *Display Current Configuration* earlier in this chapter for a full explanation of `dagconfig` outputs.

```
dagconfig -d0 reset default pos oc12 vc4c sfppwr slen=2048
```

Note position in command line.

VC-4-4c

Turns the optics modules on if they have previously been turned off.

Recommended slen value for PoS networks

```
Firmware: edag71spci_oc12c_pci_v2_23 2vp30ff1152
2006/02/21 08:53:53 (user) Serial : 8000073

SFP A: nolaser detect signal sfppwr
SFP B: nolaser detect signal sfppwr
SFP C: nolaser detect nosignal sfppwr
SFP D: nolaser detect nosignal sfppwr

Port status
Port A: lock oc12 core_on nofifo_error master enablea
Port B: lock oc12 core_on nofifo_error master enableb
Port C: nolock oc12 core_on nofifo_error master enablec
Port D: nolock oc12 core_on nofifo_error master enabled

SONET/SDH status
SONET A: oc12 vc4c scramble tu11 async
SONET B: oc12 vc4c scramble tu11 async
SONET C: oc12 vc4c scramble tu11 async
SONET D: oc12 vc4c scramble tu11 async
```

Not applicable to PoS

PoS Mode (cont.)

```
E1/T1 status
E1/T1 A: no_payload notxais
E1/T1 B: no_payload notxais
E1/T1 C: no_payload notxais
E1/T1 D: no_payload notxais
```

Not applicable to concatenated networks

```
Phy status (AMCC1213):
noeql nofcl

Concatenated Demapper Status:
pscramble crc32 pos noaidle
pscramble crc32 pos noaidle
pscramble crc32 pos noaidle
pscramble crc32 pos noaidle
```

Not applicable to PoS

```
Concatenated Mapper Status:
pscramble crc16 pos
pscramble crc16 pos
pscramble crc16 pos
pscramble crc16 pos
GPP:
varlen slen=2048 align64

PCI Burst Manager
33Mhz buffer size = 128 rx_streams = 1 tx_streams = 1
mem=112:16
TERF:
No TERF
```

Interface Statistics

When you have configured the card according to your specific requirements you can view the interface statistics to check the status of each of the links using:

```
dagconfig -d0 -si
```

Display statistics once (-s) or at 1 sec intervals (-si)

Example outputs are shown below:

> **Note:** “1” indicates the condition is present on the link “0” indicates the condition is not present on the link. See *Status Conditions* later in this chapter for a full description of each of the status conditions.

Interface Statistics (cont.)

PoS OC12 Stream

In the example below the card is locked to a PoS OC-12 stream on ports A and B, ports C and D are unused:

Line Interface Unit (LIU) conditions

Pointer conditions

```
Port   VC    LOS    LOF    OOF    B1    B2    B3    REI    C2    PTR
 A:     0     0      0      0      0     0     0     0     16   valid
 B:     0     0      0      0      0     0     0     0     16   valid
 C:     0     1      1      1      0     0     0     0     e8    lop
 D:     0     1      1      1      0     0     0     0     e8    lop
```

In the example below the card is set to OC-3 PoS while the line carries OC-12 PoS:

Line Interface Unit (LIU) conditions

Pointer conditions

```
Port   VC    LOS    LOF    OOF    B1    B2    B3    REI    C2    PTR
 A:     0     1      1      1      0     0     0     0     6d    lop
 B:     0     1      1      1      0     0     0     0     fd    lop

Port   VC    LOS    LOF    OOF    B1    B2    B3    REI    C2    PTR
 A:     0     1      1      1      0     0     0     0     44    lop
 B:     0     1      1      1      0     0     0     0     23    lop
```

ATM Cell Stream

In the example below the card is locked to an ATM OC-12 stream on ports A and B, ports C and D are unused:

Line Interface Unit (LIU) conditions

Pointer conditions

```
Port   VC    LOS    LOF    OOF    B1    B2    B3    REI    C2    PTR
 A:     0     0      0      0      0     0     0     0     13   valid
 B:     0     0      0      0      0     0     0     0     13   valid
 C:     0     1      1      1      0     0     0     0     4b    lop
 D:     0     1      1      1      0     0     0     0     e8    lop
```

Interface Statistics (cont.)

Status Conditions

A definition of each of the status conditions is described below:

Condition	Definition
`B1, B2, B3`	Bit interleaved parity error as reported by SONET B1, B2 and B3 overhead octets. Indicates the connection between the card and the network is impaired. If `OOF` and `LOF` conditions are also set the OCx carrier configuration is incorrect. If `OOF` and `LOF` conditions are not set the there is a signal problem related to low light levels reaching the optical receivers, or there are true SONET-level errors on the equipment operating the link.
`C2`	Path signal label. Reflects the content of SONET C2 overhead octet. Typical settings are: 13ATM 16 PPP w/SPE scrambling CF PPP wo/SPE scrambling Changing values for this condition indicate a SONET level error.
`LOF`	Loss of Frame. Indicates Out of Frame (`OOF`) condition has been asserted for more than 2 milliseconds.
`PTR`	Indicates if the pointer processing logic has locked to the SONET stream. Possible values are: `valid`: The pointer has locked to the SONET stream. `lop` (loss of pointer): The pointer has not locked to the SONET stream and may indicate incorrect OC-3/OC-12 setting. `conc`: The card is configured for channelised data but the network is concatenated.
`LOS`	Loss of Signal. There is either no signal at the receiver or the optical signal strength is too low for the card to recognize.
`OOF`	Out of Frame. The section overhead processor is not locked to the SONET stream and may indicate incorrect OC-3/OC-12 setting.
`REI`	Remote Error Indicator.
`VC`	Virtual container number. Should always be “`0`” for concatenated networks.

ATM Network

If the card is connected to an ATM network and you have loaded the correct firmware for the network configuration the following values apply:

- C2 should be “`13`”
- PTR should be “`valid`”
- Remaining statistics should be “`0`”

PoS Network

If the card is connected to an PoS network and you have loaded the correct firmware for the network configuration the following values apply:

- C2 should be "16" for PPP or "cf" for HDLC
- PTR should be "valid"
- Remaining statistics should be "0"

Verify Configuration

You can verify the card configuration by checking settings and path label for any errors as follows:

- Ensure LOS is 0, otherwise check light levels.
- Ensure OOF and LOF are 0, otherwise change OC-3 settings to OC-12 or vice versa.
- Ensure PTR is valid
- Ensure no bit interleaved parity errors occur, otherwise check cabling and light levels.
- Ensure C2 is correct for the payload
- Ensure PoS scrambling and CRC settings are correct.

Chapter 5: Channelised Configuration

Load the FPGA Images

Note: Before you can configure the card for capture you must first load the card with the appropriate FPGA images for the type of data you want to capture. It is important you understand the protocol used by the network to which you want to connect. If you do not load the correct image for the protocol you will not be able to capture data.

- Place the card in "`eql`" mode to prevent any erroneous signals interfering with the images during loading using

  `dagconf –d0 eql` (where "`0`" is the device number of the DAG card)

- PCI-Express FPGA image:

  ```
  dagrom -rvp –d0 -f xilinx/dag71spci-chan-terf.bit
  ```

- PHY FPGA image:

  ```
  dagld –x –d0 xilinx/dag71spp-chan-terf.bit:
  xilinx/dag71spp-chan-terf.bit
  ```

Available Configurations

The available channelised configurations are shown below:

Number of Ports	Line Type	VC Type and Number	Protocol
4	OC-3/STM-1	252 x VC-12 (E1) 336 x VC-11 (T1)	ATM/Bit-HDLC/ RAW
2	OC-12/STM-4	504 x VC-12 (E1) 672 x VC-11 (T1)	ATM/Bit-HDLC/ RAW

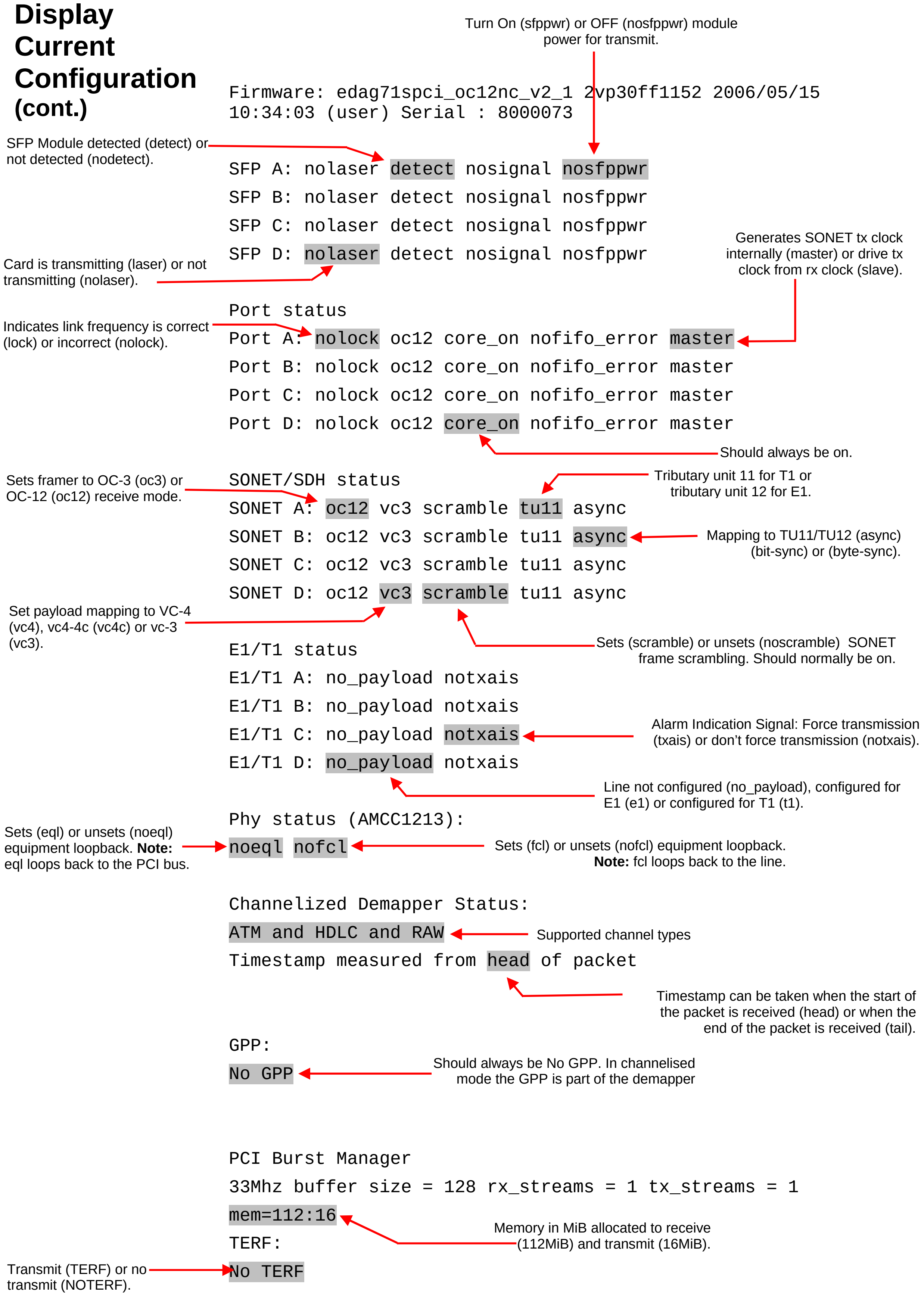
Display Current Configuration

Once you have loaded the appropriate images you should run the `dagconfig` tool without arguments to display the current card configuration and verify the firmware has loaded correctly, using:

`dagconfig –d0` (where "`0`" is the device number of the DAG card)

An explanation of an the default `dagconfig` output is shown on the following page:

Display Current Configuration (cont.)



```
Firmware: edag71spci_oc12nc_v2_1 2vp30ff1152 2006/05/15
10:34:03 (user) Serial : 8000073

SFP A: nolaser detect nosignal nosfppwr
SFP B: nolaser detect nosignal nosfppwr
SFP C: nolaser detect nosignal nosfppwr
SFP D: nolaser detect nosignal nosfppwr

Port status
Port A: nolock oc12 core_on nofifo_error master
Port B: nolock oc12 core_on nofifo_error master
Port C: nolock oc12 core_on nofifo_error master
Port D: nolock oc12 core_on nofifo_error master

SONET/SDH status
SONET A: oc12 vc3 scramble tu11 async
SONET B: oc12 vc3 scramble tu11 async
SONET C: oc12 vc3 scramble tu11 async
SONET D: oc12 vc3 scramble tu11 async

E1/T1 status
E1/T1 A: no_payload notxais
E1/T1 B: no_payload notxais
E1/T1 C: no_payload notxais
E1/T1 D: no_payload notxais

Phy status (AMCC1213):
noeql nofcl

Channelized Demapper Status:
ATM and HDLC and RAW
Timestamp measured from head of packet

GPP:
No GPP


PCI Burst Manager
33Mhz buffer size = 128 rx_streams = 1 tx_streams = 1
mem=112:16
TERF:
No TERF
```

Turn On (sfppwr) or OFF (nosfppwr) module power for transmit.

SFP Module detected (detect) or not detected (nodetect).

Card is transmitting (laser) or not transmitting (nolaser).

Generates SONET tx clock internally (master) or drive tx clock from rx clock (slave).

Indicates link frequency is correct (lock) or incorrect (nolock).

Should always be on.

Sets framer to OC-3 (oc3) or OC-12 (oc12) receive mode.

Tributary unit 11 for T1 or tributary unit 12 for E1.

Mapping to TU11/TU12 (async) (bit-sync) or (byte-sync).

Set payload mapping to VC-4 (vc4), vc4-4c (vc4c) or vc-3 (vc3).

Sets (scramble) or unsets (noscramble) SONET frame scrambling. Should normally be on.

Alarm Indication Signal: Force transmission (txais) or don't force transmission (notxais).

Line not configured (no_payload), configured for E1 (e1) or configured for T1 (t1).

Sets (eql) or unsets (noeql) equipment loopback. **Note:** eql loops back to the PCI bus.

Sets (fcl) or unsets (nofcl) equipment loopback. **Note:** fcl loops back to the line.

Supported channel types

Timestamp can be taken when the start of the packet is received (head) or when the end of the packet is received (tail).

Should always be No GPP. In channelised mode the GPP is part of the demapper

Memory in MiB allocated to receive (112MiB) and transmit (16MiB).

Transmit (TERF) or no transmit (NOTERF).

Display Current Configuration (cont.)

If the firmware has not loaded correctly the `dagconfig` output will indicate "`component not found`" as the `SONET/SDH status` and `E1/T1 status` as shown below:

```
SONET/SDH status
SONET A:  component not found.
SONET B:  component not found.
SONET C:  component not found.
SONET D:  component not found.

E1/T1 status
E1/T1 A:  component not found.
E1/T1 B:  component not found.
E1/T1 C:  component not found.
E1/T1 D:  component not found.
```

In this case you should check the firmware image that you are using and ensure that it is correct for your network configuration and protocol. Then reload the firmware as described in *Load FPGA Images* earlier in this chapter.

If after performing these steps `dagconfig` still displays "`component not found`" please contact Endace Customer Support at support@endace.com for further assistance.

Verify Optical Signal

As shown on the previous page, the `dagconfig` output displays whether or not a port is receiving a valid optical signal.

Valid signal received

```
SFP A: laser detect signal nosfppwr
SFP B: laser detect nosignal nosfppwr
```

No valid signal received

To receive a valid optical signal, both ends of the link must use be using the same type of optical transceivers. In addition the optical fiber used must match the requirements of the optical transceiver and be in good condition.

Verify Mapping/ Framing Setup

Parameters

As shown on the previous page, the `dagconfig` output displays whether or not mapping and framing is setup correctly.

For correct setup both ends of the link must match the framing and mapping parameters. The main parameters involved in setting the framing are:

- Link speed (`oc3/oc12`)
- Clock master (`master/slave`)
- Payload mapping (`vc4/vc3`)

Link setup correctly

```
Port status
Port A: lock oc12 core_on nofifo_error slave
Port B: nolock oc12 core_on nofifo_error slave
```

Link not setup correctly

```
SONET/SDH status
SONET A: oc12 vc4c scramble tu11 async
SONET B: oc12 vc4c scramble tu11 async
SONET C: oc12 vc4c scramble tu11 async
SONET D: oc12 vc4c scramble tu11 async
```

Different Port Configurations

The DAG card allows different ports to have different configurations. For example if you want to configure only Ports A and B for OC-3c VC-4c you could do so using the following command:

```
dagconfig –d0 default -1 -4 oc3 vc4

Port status
Port A: nolock oc3 core_on nofifo_error master
Port B: nolock oc3 core_on nofifo_error master
Port C: nolock oc12 core_on nofifo_error master
Port D: nolock oc12 core_on nofifo_error master

SONET/SDH status
SONET A: oc3 vc4 scramble tu11 async
SONET B: oc3 vc4 scramble tu11 async
SONET C: oc12 vc4 scramble tu11 async
SONET D: oc12 vc4 scramble tu11 async
```

Configure Line Type

Before you configure the DAG 7.1S for specific ATM, Bit HDLC or RAW data capture you must configure the card for the correct line type and speed.

Please refer to *Display Current Configuration* earlier in this chapter for a full explanation of `dagconfig` outputs.

Note: The order in which the configuration options appear in the command line is important. Therefore you should use `default` and `reset` before other options in the command line.

The example below has been configured for OC-12 VC-4 on ports A and B with ports C and D unused:

```
      dagconfig -d0 reset default laser sfppwr OC12 vc4 tu11 t1
```

Note position in command line

Turns optics modules on if they have previously been turned off.

VC-4

```
Firmware: edag71spci_oc12nc_v2_1 2vp30ff1152 2006/05/15
10:34:03 (user) Serial : 8000073

SFP A: laser detect signal sfppwr
SFP B: laser detect signal sfppwr
SFP C: laser detect nosignal nosfppwr
SFP D: laser detect nosignal nosfppwr

Port status
Port A: lock oc12 core_on nofifo_error master
Port B: lock oc12 core_on nofifo_error master
Port C: nolock oc12 core_on nofifo_error master
Port D: nolock oc12 core_on nofifo_error master

SONET/SDH status
SONET A: oc12 vc4 scramble tu11 async
SONET B: oc12 vc4 scramble tu11 async
SONET C: oc12 vc4 scramble tu11 async
SONET D: oc12 vc4 scramble tu11 async

E1/T1 status
E1/T1 A: t1 notxais
E1/T1 B: t1 notxais
E1/T1 C: no_payload notxais
E1/T1 D: no_payload notxais

Phy status (AMCC1213):
noeql nofcl

Channelized Demapper Status:
ATM and HDLC and RAW
Timestamp measured from head of packet

GPP:
No GPP

PCI Burst Manager
33Mhz buffer size = 128 rx_streams = 1 tx_streams = 1
mem=112:16
TERF:
No TERF
```

Interface Statistics

When you have configured the card according to your specific line type and speed requirements you can view the interface statistics to check the status of each of the links using:

```
dagconfig -d0 -si
```

Display statistics once (-s)
or at 1 sec intervals (-si)

Example outputs are shown below:

> **Note:** "`1`" indicates the condition is present on the link "`0`" indicates the condition is not present on the link. See *Status Conditions* later in this chapter for a full description of each of the status conditions.

E1 OC-12

In the example below the card is set to `OC-12 vc4 tu12 e1` on all ports. The card is not receiving any traffic:

Line Interface Unit (LIU) conditions

Pointer conditions

```
Port   VC   LOS   LOF   OOF   B1   B2   B3   REI   C2   PTR
 A     0     1     1     1    0    0    0     0    e8   lop
       1    ..    ..    ..   ..   ..    1     0    63   lop
       2    ..    ..    ..   ..   ..    1     1    06   lop
       3    ..    ..    ..   ..   ..    1     0    ca   lop
 B     0     1     1     1    0    0    0     1    e8   lop
       1    ..    ..    ..   ..   ..    1     0    63   lop
       2    ..    ..    ..   ..   ..    1     0    2a   lop
       3    ..    ..    ..   ..   ..    1     0    99   lop
 C     0     1     1     1    0    0    1     0    e8   lop
       1    ..    ..    ..   ..   ..    1     0    63   lop
       2    ..    ..    ..   ..   ..    1     1    2a   lop
       3    ..    ..    ..   ..   ..    1     0    9d   lop
 D     0     1     1     1    0    0    0     1    e8   lop
       1    ..    ..    ..   ..   ..    1     1    63   lop
       2    ..    ..    ..   ..   ..    1     0    2a   lop
       3    ..    ..    ..   ..   ..    1     0    9d   lop
```

T1 OC-12

In the example below the card is set to `oc12 vc3 tu11 t1` on all ports:

Line Interface Unit (LIU) conditions

Pointer conditions

```
Port   VC   LOS   LOF   OOF   B1   B2   B3   REI   C2   PTR
 A      0    0     0     0     0    0    0    0    02   valid
        1   ..    ..    ..    ..   ..    0    0    02   valid
        2   ..    ..    ..    ..   ..    0    0    02   valid
        3   ..    ..    ..    ..   ..    0    1    02   valid
        4   ..    ..    ..    ..   ..    0    1    02   valid
        5   ..    ..    ..    ..   ..    0    0    02   valid
        6   ..    ..    ..    ..   ..    0    0    02   valid
        7   ..    ..    ..    ..   ..    0    1    02   valid
        8   ..    ..    ..    ..   ..    0    0    02   valid
        9   ..    ..    ..    ..   ..    0    1    02   valid
       10   ..    ..    ..    ..   ..    0    1    02   valid
       11   ..    ..    ..    ..   ..    0    1    02   valid
 B      0    1     1     1     0    0    0    0    02   valid
        1   ..    ..    ..    ..   ..    1    1    02   valid
        2   ..    ..    ..    ..   ..    1    1    02   valid
        3   ..    ..    ..    ..   ..    1    0    02   valid
        4   ..    ..    ..    ..   ..    1    1    02   valid
        5   ..    ..    ..    ..   ..    1    1    02   valid
        6   ..    ..    ..    ..   ..    1    0    02   valid
        7   ..    ..    ..    ..   ..    1    0    02   valid
        8   ..    ..    ..    ..   ..    1    1    02   valid
        9   ..    ..    ..    ..   ..    1    1    02   valid
       10   ..    ..    ..    ..   ..    1    1    02   valid
       11   ..    ..    ..    ..   ..    1    0    02   valid
 C      0    1     1     1     0    1    0    0    02   valid
        1   ..    ..    ..    ..   ..    1    1    02   valid
        2   ..    ..    ..    ..   ..    1    0    02   valid
        3   ..    ..    ..    ..   ..    1    0    02   valid
        4   ..    ..    ..    ..   ..    1    0    02   valid
        5   ..    ..    ..    ..   ..    1    1    02   valid
        6   ..    ..    ..    ..   ..    1    1    02   valid
        7   ..    ..    ..    ..   ..    1    1    02   valid
        8   ..    ..    ..    ..   ..    1    1    02   valid
        9   ..    ..    ..    ..   ..    1    0    02   valid
       10   ..    ..    ..    ..   ..    1    0    02   valid
       11   ..    ..    ..    ..   ..    1    0    02   valid
 D      0    1     1     1     0    0    1    0    02   valid
        1   ..    ..    ..    ..   ..    1    1    02   valid
        2   ..    ..    ..    ..   ..    1    1    02   valid
        3   ..    ..    ..    ..   ..    1    0    02   valid
        4   ..    ..    ..    ..   ..    1    1    02   valid
        5   ..    ..    ..    ..   ..    1    1    02   valid
        6   ..    ..    ..    ..   ..    1    0    02   valid
        7   ..    ..    ..    ..   ..    1    1    02   valid
        8   ..    ..    ..    ..   ..    1    1    02   valid
        9   ..    ..    ..    ..   ..    1    1    02   valid
       10   ..    ..    ..    ..   ..    1    0    02   valid
       11   ..    ..    ..    ..   ..    1    0    02   valid
```

Interface Statistics (cont.

Status Conditions

A definition of each of the status conditions is described below:

Condition	Definition
`B1, B2, B3`	Bit interleaved parity error as reported by SONET B1, B2 and B3 overhead octets. Indicates the connection between the card and the network is impaired. If `OOF` and `LOF` conditions are <u>also</u> set the OCx carrier configuration is incorrect. If `OOF` and `LOF` conditions are <u>not</u> set the there is a signal problem related to low light levels reaching the optical receivers, or there are true SONET-level errors on the equipment operating the link.
`C2`	Path signal label. Reflects the content of SONET C2 overhead octet. Typical settings are: • If `PTR` is `valid` – "`02`" • If `PTR` is `AIS` – "`FF`" • If `PTR` is `conc` – "`13`", "`16`" or "`cf`" • If `PTR` is `lop` – any value Changing values for this condition indicate a SONET level error.
`LOF`	Loss of Frame. Indicates Out of Frame (`OOF`) condition has been asserted for more than 2 milliseconds.
`PTR`	Indicates if the pointer processing logic has locked to the SONET stream. Possible values are: `valid`: The pointer has locked to the SONET stream. `lop` (loss of pointer): The pointer has not locked to the SONET stream and may indicate incorrect OC-3/OC-12 setting. `conc`: The line is configured for concatenated data but the network is channelised.
`LOS`	Loss of Signal. There is either no signal at the receiver or the optical signal strength is too low for the card to recognize.
`OOF`	Out of Frame. The section overhead processor is not locked to the SONET stream and may indicate incorrect OC-3/OC-12 setting.
`REI`	Remote Error Indicator.
`VC`	Virtual container number. Should always be "`0`" for concatenated networks.

Verify Configuration

If the card is connected to a network and you have loaded the correct firmware for the network configuration the following values apply:

- C2 should be "`02`"
- PTR should be "`valid`"
- Remaining statistics should be "`0`"

You can verify the card configuration by checking settings and path label for any errors as follows:

- Ensure `LOS` is 0, otherwise check light levels.
- Ensure `OOF` and `LOF` are zero, otherwise change OC-3 settings to OC-12 or vice versa.
- Ensure no bit interleaved parity errors occur, otherwise check cabling and light levels.
- Ensure `C2` is correct for the payload.
- Ensure ATM `LCD` is off and `SYNC` set.

Ensure PoS scrambling and CRC settings are correct.

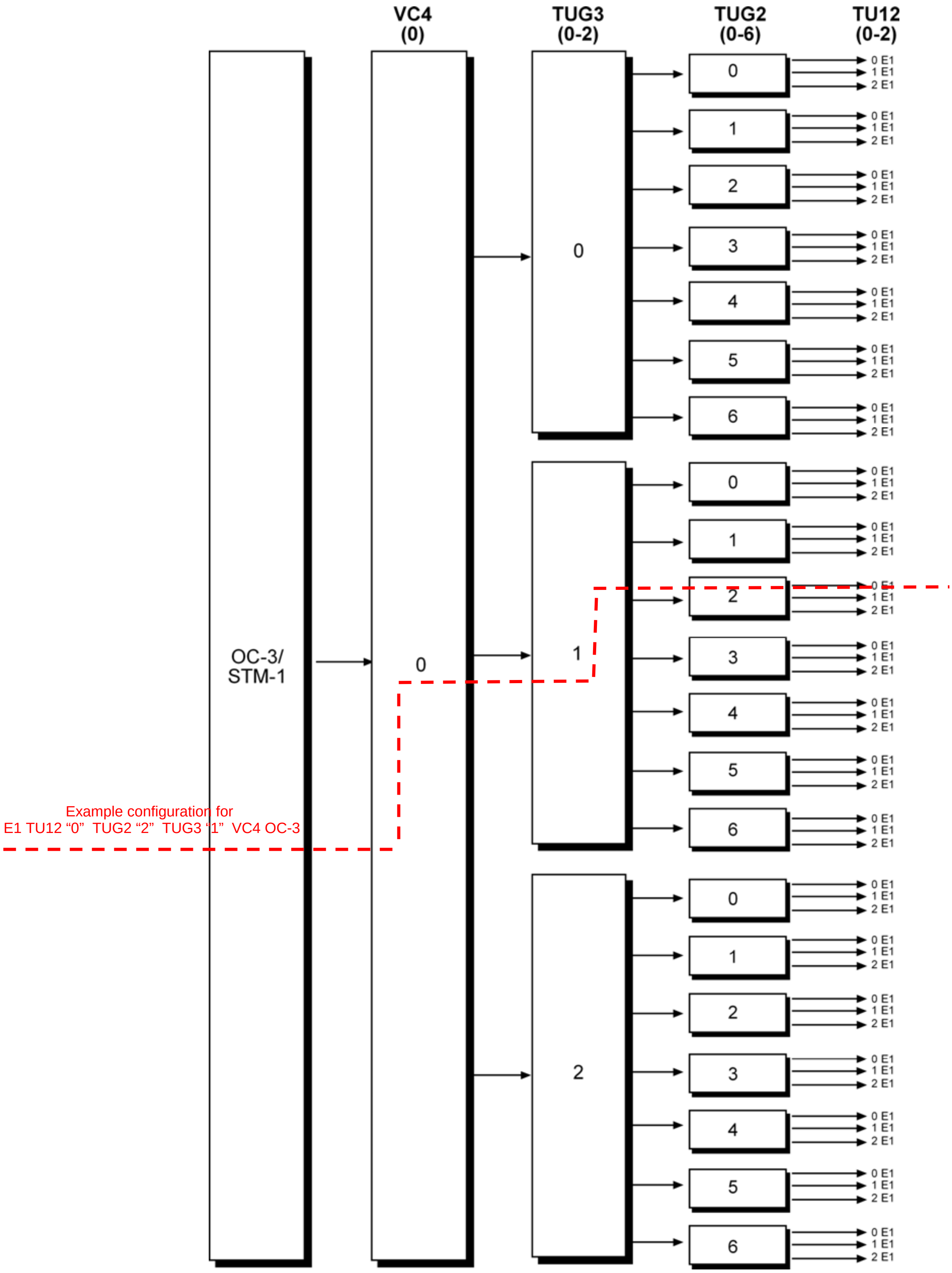
Configuring Channels

Valid Configurations

E1 OC-3/STM-1 and OC-12/STM-4

There are 63 valid channel configuration options for E1 OC-3/STM-1 and these are shown in the diagram below.

For E1 OC-12/STM-4 there are 252 (4 x 63) valid channel configuration options because OC-12/STM-4 supports 4 x VC4.



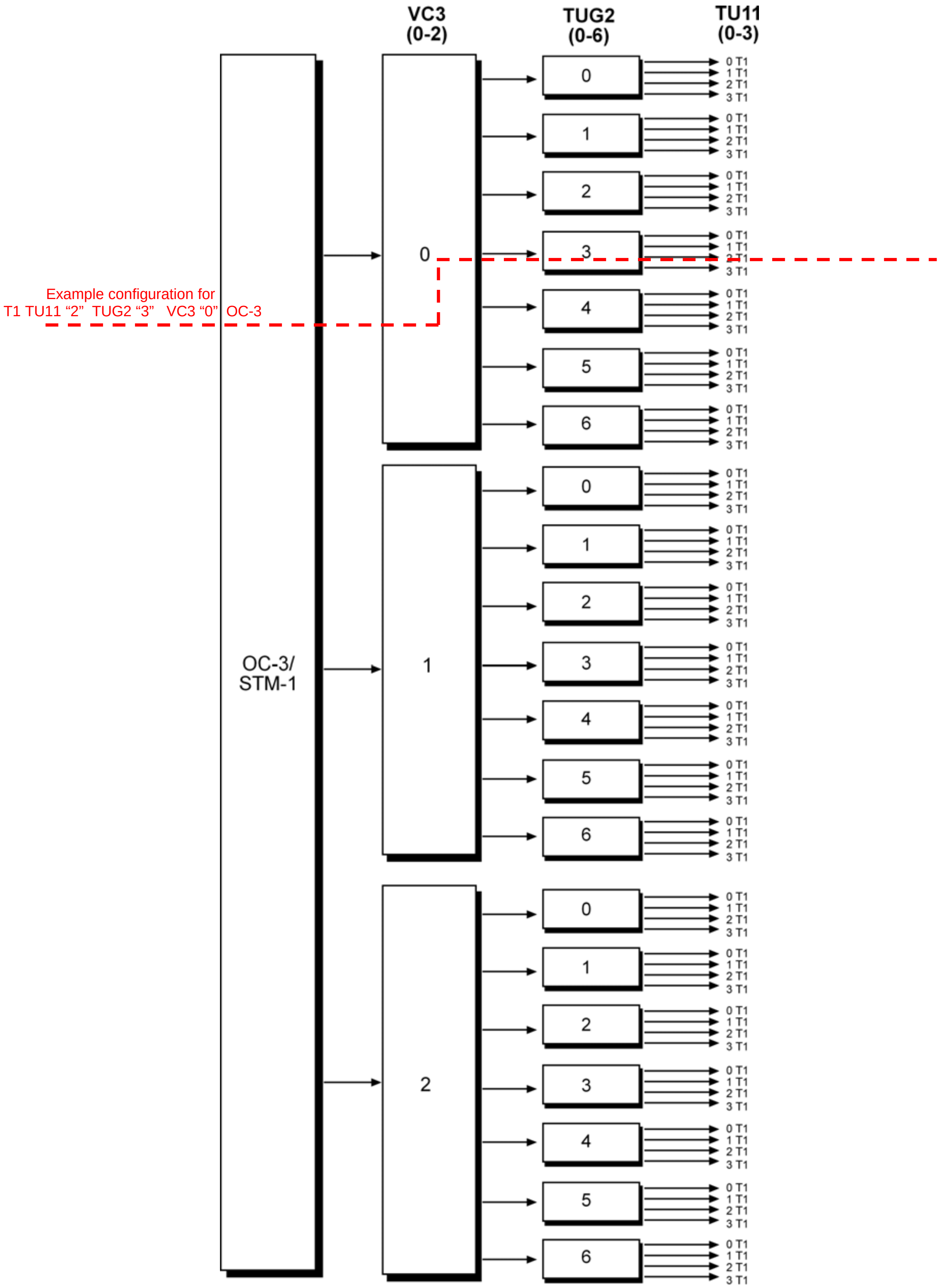
Example configuration for
E1 TU12 “0” TUG2 “2” TUG3 “1” VC4 OC-3

Configuring Channels (cont.)

T1 OC-3/STM-1 and OC-12/STM-4

There are 84 valid channel configuration options for T1 OC-3/STM-1 and these are shown in the diagram below..

For T1 OC-12/STM-4 there are 336 (4 x 84) valid channel configuration options because T1 OC-12/STM-4 supports 12 x VC3.



Example configuration for
T1 TU11 “2” TUG2 “3” VC3 “0” OC-3

Configuring Channels (cont.)

Supported Channel Types

The DAG 7.1S supports capture of ATM, Bit HDLC and RAW data over the the following channel types:

Channel Type	Usage
PCM 24	E1: Timeslots `1-24` (25-31 unused) T1: Timeslots `1-24` (all available slots)
PCM 30	E1:Timeslots `1-15` and `17-31` (`0` and `16` used for framing and signalling) T1: Timeslots 1-15 and 17-24 (`16` used for signalling)
PCM 31	E1: Timeslots `1-31` (`0` is used for framing) T1: Timeslots 1-24 (all available slots)

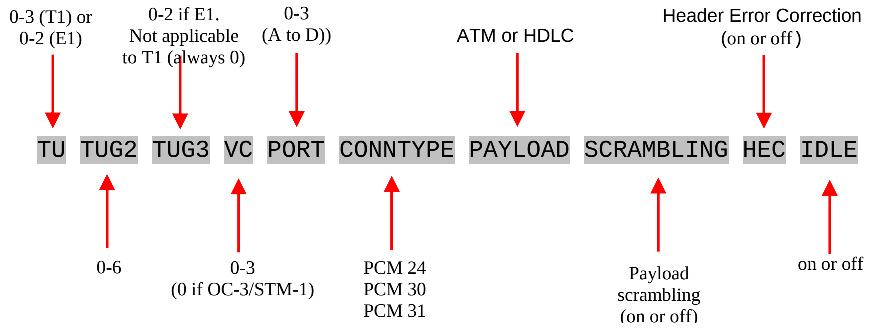
Configuration File

You can configure a specific channel type for capture of ATM, Bit HDLC or RAW data using `dagconfig` to read a channel configuration file and then create each of the channels defined in that file. You can create a channel configuration file using any common text editor such as Notepad, VI Editor, Emacs etc.

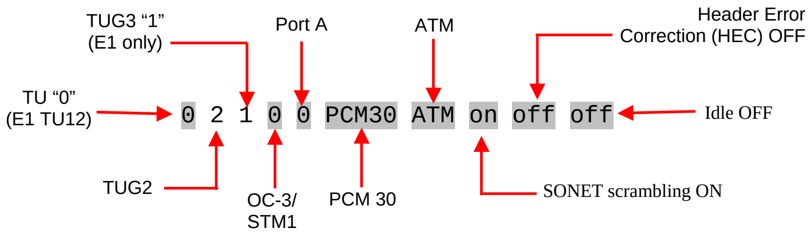
Note: The file must be a plain ASCII text file

Each line in the configuration file represents the settings for an individual channel allowing you to configure multiple channels using the one configuration file.

The file is described below:



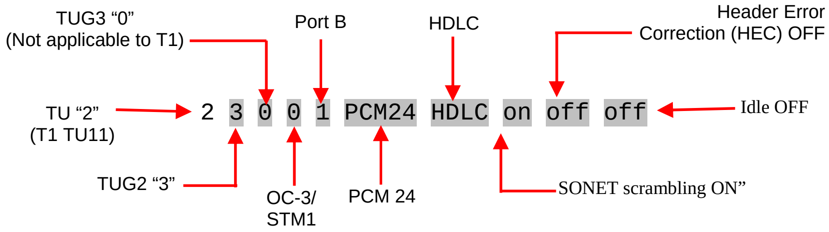
In the example line below the channel is configured for E1 channelised OC-3/STM-1, port A containing ATM over PCM-30.



This configuration is shown in the *E1 OC-3/STM-1* diagram in *Valid Configurations* earlier in this chapter.

Configuring Channels (cont.)

In the example line below the channel is configured for T1 channelised OC-3/STM-1, port B containing HDLC over PCM-24.



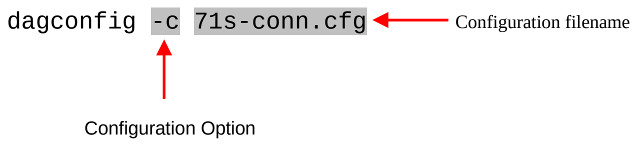
```
2 3 0 0 1 PCM24 HDLC on off off
```

This configuration is shown in the *T1 OC-3/STM-1* diagram in *Valid Configurations* earlier in this chapter.

An example configuration file containing the settings for 5 channels is shown below:

```
#Channel 1:
0 0 0 0 0 PCM30 ATM on off off
#Channel 2:
1 0 0 0 0 PCM30 ATM on off off
#Channel 3:
2 0 0 0 0 PCM30 ATM on off off
#Channel 4:
0 1 0 0 0 PCM30 ATM on off off
#Channel 5:
1 1 0 0 0 PCM30 ATM on off off
```

To configure channels with the configuration file use:



```
dagconfig -c 71s-conn.cfg
```

Chapter 6: Capturing Data

Starting a Session

For a typical data capture session follow the steps listed below:

- Move to the dag directory,
- Load the appropriate driver,
- Then load the appropriate FPGA image to each DAG card as described in *Load the FPGA Images* in *Chapter 4: Concatenated Configuration* and *Chapter 5: Channelised Configuration* earlier in this User Guide.
- Set the integrity of the card's physical layer and check the integrity of the physical layer to each DAG card. For example:

```
dagconfig –d0 default
```

- Start the capture session using:

```
tools/dagsnap -d0 –v -o tracefile
```

> **Note:** You can use the `-v` option to provide user information during a capture session although you may want to omit it for automated trace runs.

By default `dagsnap` will run indefinitely but can be stopped using *CTRL+C*. You can also configure `dagsnap` to run for a fixed time period then exit.

Setting Captured Packet Size

Snaplength

Before you begin to capture data you can set the size that you want the captured packets to be. You can do this using the `dagconfig` tool to define the packet snaplength (`slen`).

> **Note:** The snaplength value must be a multiple of 8 and in the range 16 to 2040 inclusive.

By default `slen` which is the portion of the packet that you want to capture, is set to 48. This means that only the first 48 bytes of each packet will be captured.

If for example you want to capture only the IP header of each packet you may want to set the length to a smaller value. Alternatively if you want to ensure you capture the whole packet you can set the length to a larger value.

> **Note:** The ERF header is not included in the `slen` value. Therefore a `slen` of 48 will produce a capture record of 48 bytes plus the number of bytes in the ERF header.

Setting Captured Packet Size (cont.)

Variable/Fixed Length

The DAG card is able to capture packets in two ways. They are:

- Variable length capture (`varlen`),
- Fixed length capture (`novarlen`).

In variable length (`varlen`) mode the card will capture the whole packet, providing its size is less than the `slen` value. Therefore to use this capture mode effectively you should set the `slen` value to the largest number of bytes that a captured packet is likely to contain.

Any packet that is larger than the `slen` value will be truncated to that size. Any packet that is smaller than the `slen` value will be captured at its actual size therefore producing a shorter record which save bandwidth and storage space.

The example below shows a configuration for variable length full packet capture:

```
dagconfig -d0 varlen slen=2040
```

Maximum allowable slen value

Variable length capture

In fixed length (`novarlen`) mode the card will capture all packets at he same length. Any packet that is longer than the `slen` value will be truncated to that size, in the same way as for `varlen` capture. However any packet that is shorter than the `slen` value will be captured at its full size and then padded out to the size of the `slen` value.

This means that in `novarlen` mode you should avoid large `slen` values because short packets arriving will produce records with a large amount of padding which wastes bandwidth and storage space.

The example below shows a configuration for fixed length packet capture that will produce a 64-byte record (48 byte payload plus 16 byte ERF header):

```
dagconfig -d0 novarlen slen=48
```

First 48 bytes will be captured

Fixed length capture.
Note: This option is not available in channelised mode.

Enabling/ Disabling Ports

You can also enable and disable individual ports for capture using the `dagconfig` tool as shown below:

```
dagconfig -d0 disable
```

Note: DAG ports are enabled by default. You do not need to use `dagconfig` to enable the card in order to begin capture unless you have previously disabled it.

High Load Performance

Overview

As the DAG 7.1S card captures packets from the network link, it writes a record for each packet into a large buffer in the host PC's main memory.

Avoiding Packet Loss

To avoid packet loss, the user application reading the record, such as `dagsnap`, must be able to read records out of the buffer faster than they arrive. If not the buffer will eventually fill and packet records will be lost.

If the user process is writing records to hard disk, it may be necessary to use a faster disk or disk array. If records are being processed in real-time, a faster host CPU may be required.

In Linux and Free BSD, when the PC buffer fills, the following message displays on the PC screen:

```
      kernel: dagN: pbm safety net reached 0xNNNNNNNN
```

The same message is also printed to log /var/log/messages. In addition, when the PC buffer fills the "Data Capture" LED on the card will flash or flicker, or may go OFF completely.

In Windows no screen message displays to indicate when the buffer is full. Please contact Endace Customer Support at support@endace.com for further information on detecting buffer overflow and packet loss in Windows

Detecting Packet Losses

Once the buffer fills, any new packets arriving will be discarded by the DAG card until some data is read out of the buffer to create free space.

You can detect any such losses by observing the Loss Counter (`lctr` field) of the Extensible Record Format [ERF]. See *Chapter 8: Data Formats* later in this User Guide for more information on the Endace ERF.

Increasing Buffer Size

You can increase the size of the host PC buffer to enable it to cope with bursts of high traffic load on the network link.

By default the `dagmem` driver reserves 32MB of memory per DAG card in the system. However if you are capturing at OC-12/STM-4 (622Mbps) rates or above, you may require a larger buffer.

For Linux/BSD 128MB or more is recommended. However you can change the amount of reserved memory by editing the file `/etc/modules` as follows

```
# For DAG 3.x, default 32MB/card
dagmem
#
# For DAG 4.x or 6.x, use more memory per card,
E.G.
# dagmem dsize=128m
```

For Windows the upper limit is 32MB. This is usually sufficient however if you do need to increase the amount of reserved memory please contact Endace customer support at support@endace.com for more information

The `dsize` option sets the amount of memory used per DAG card in the system.

> **Note:** The value of dsize multiplied by the number of DAG cards in the system must be less than the amount of physical memory installed, as well as less than 890MB.

Transmitting

Configuration

The DAG 7.1S is able to transmit as well as receive packets and can capture received traffic while transmitting. This allows you to use capture tools such as `dagsnap`, `dagconvert`, and `dagbits` while `dagflood` is sending packets.

To configure the DAG card for transmission, you must allocate some memory to a transmit stream.

In the `dagconfig` output, `buf=nMB` indicates that `n` mebibytes of memory have been allocated to the DAG card in total. You can split this allocation between the receive and transmit stream buffers. The split is displayed as a ratio as shown below:

`mem=X:Y`, where

`X` is the memory allocated in MB to the rx stream “0”
`Y` is the memory allocated in MB to tx stream “1” in MB.

By default, the memory allocation is evenly split between the rx streams, with the transmit streams having no memory allocated.

Transmitting (cont.)

If you wish to use the card for both transmitting and receiving, you can use the `rxtx` option. This allocates 16MiB of memory to each transmit stream, and divides the remaining memory between the receive streams.

Alternatively you can set the memory allocation directly using the `mem= X:Y` option.

! **Note:** You can not change the stream memory allocations while packet capture or transmission is in progress.

Explicit Packet Transmission

The operating system does not recognize the DAG card as a network interface and will not respond to ARP, ping, or router discovery protocols.

The DAG card will only transmit packets that are explicitly provided by the user. This allows you to use the DAG card as a simple traffic load generator.

You can also use it to retransmit previously recorded packet traces. The packet trace is transmitted at 100% line rate. The packet timing of the original trace file is not reproduced.

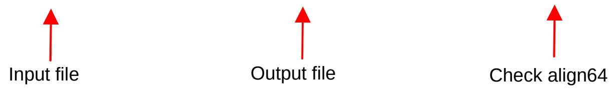
Trace Files

You can use the `dagflood` tool to transmit ERF format trace files, providing the files contain only ERF records of the type matching the current link configuration.

In addition the length of the ERF records to be transmitted must be a multiple of 64-bits. You can configure this when capturing packets for later transmission by setting 64-bit alignment using the `dagconfig align64` command.

If packets have been captured without using the `align64` option you can convert the trace files so that they can be transmitted by using `dagconvert` as shown below:

```
dagconvert -v -i tracefile.erf -o tracefile.erf -A8
```



Alternatively if you are unsure if a trace file is 64-bit aligned you can test the file using `dagbits` as shown below:

```
dagbits -v align64 -f tracefile.erf
```

If you do not have any ERF trace files available, you can use `daggen` to generate trace files containing simple traffic patterns. This allows the DAG card to be used as a test traffic generator.

For further information on using `daggen` please refer to the *EDM04-06: Daggen User Guide* available from Endace Customer Support at support@endace.com.

Configuring Extended Functions

Overview

The embedded IXP network processor provides the means to perform the following extended functions:

- Reassembly of AAL2/AAL5 frames
- IP filtering of PoS packets

Before you can make use of these extended functions you must ensure you have completed the following steps as described in *Chapter 3: Configuring the Card* earlier in this User Guide:

- Load the channelised or concatenated firmware images depending upon the function you want to perform,
- Configure the card based upon your network settings,
- Check (using `dagsnap`) that you are able to capture ATM and either PoS or bit-HDLC as appropriate.

Loading the Images

To make use of the extended functionality provided by the IXP Network processor, you must load the appropriate IXP images. You can do this using the dagrom command line option tool and the region (–c) option. There are three image required. They are:

- Bootloader,
- Kernel, and
- Filesystem.

- Load the Bootloader image using:

```
dagrom –d0 –rvi –cb –f d71S_bootloader.rom
```

- Then load the kernel using:

```
dagrom –d0 –rvi –ck –f d71S_kernel.rom
```

- Then load the appropriate file system depending upon the function you want to use:

for AAL/2AAL5 reassembly on a concatenated network load:

```
dagrom –d0 –rvi –cf –f d71S_sar_conc_filesystem.rom
```

for AAL2/AAL5 reassembly on a channelised network load:

```
dagrom –d0 –rvi –cf –f d71S_sar_chan_filesystem.rom
```

for PoS IP filtering load:

```
dagrom –d0 –rvi –cf –f d71S_ipf_filesystem.rom
```

Configuring Extended Functions (cont.)

Starting the IXP

You can start the IXP using one of the following command line tools with the “`-x`” option depending upon the function you want to perform.

- `dagixp_filter_loader` for PoS IP filtering
- `dagsar_stub` for AAL2/AAL5 reassembly

> **Note**: You can also make a connection to the embedded processor using the `dagema` library. For more information on the `dagema` library please contact Endace Customer Support at support@endace.com.

Directing Data to the IXP

In normal circumstances when you run `dagconfig`, data is directed from the line directly to the host computer.

However to enable any of the extended functions the data must be directed from the line then to the IXP processor, then to the host computer.

Both the `dagixp_filter_loader` and `dagsar_stub` tools do this by default. Alternatively you can use the “Erf-Mux” component (`kComponent ErfMux`) provided by the Endace Configuration and Status API library. For more information please refer to the *Configuration & Status API User Guide* available from Endace Customer Support at support@endace.com.

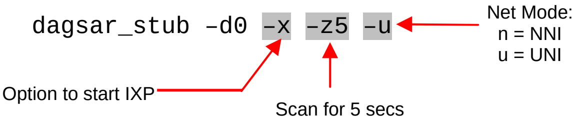
Using the AAL Reassembler

The AAL Reassembler allows you to reassemble AAL5 or AAL2 frames on the DAG 7.1S card without involving the host computer in the processing. The reassembler receives ATM traffic from the line and then sends it to the host either unchanged, dropped or reassembled into AAL5 or AAL2, frames depending upon the configuration used.

You can use different configurations on different virtual connections, allowing only the required data to be reassembled, and other data to be either preserved or rejected.

The SAR API is a library that provides the functionality to configure and read the status of the AAL Reassembler. The command line tool `dagsar_stub` provides an application layer around the SAR API and can be used to directly configure the AAL Reassembler.

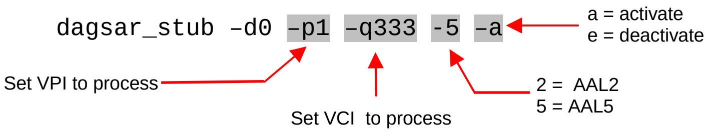
- To scan for available virtual connections on the line, use `dagsar_stub` with the “`-z`” option to specify the duration of the scan in seconds. Ensure you use the “`-x`” option to start the IXP on the first scan:



```
dagsar_stub -d0 -x -z5 -u
```

Using the AAL Reassembler (cont.)

- To configure the ATM connection to reassemble either AAL2 or AAL5 SSSAR frames, use `dagsar_stub`. Ensure that if the IXP has not previously been started, you use the "`-x`" option to do so.



```
dagsar_stub -d0 -p1 -q333 -5 -a
```

- Configure as many additional connections to reassemble AAL frames as required. You do not need to use the "`-x`" option on additional connections.
- Start capturing the data using:

```
dagsnap -d0 -v -o output_filename
```

For more information on the specific configuration options available for the AAL5/AAL2 reassembler please refer to the SAR API Programming Guide available from Endace Customer Support at support@endace.com.

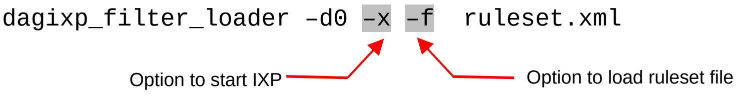
Using the PoS IP Filter

The PoS IP filter allows IPv4/IPv6 and Layer 4 filtering of PoS packets on the DAG 7.1S card without involving the host computer in processing. The filter receives PoS traffic from the line and then either sends it to the host unchanged or drops it, depending on the configuration used.

By default no filter rules are setup so all PoS traffic is sent to the host unchanged. You can setup filters using the DAGIXP Filtering API library or by using the `dagixp_filter_loader` command line tool.

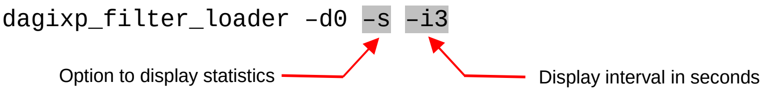
The `dagixp_filter_loader` tool accepts an XML based filter ruleset file which it then parses and loads directly into the DAG card, overwriting any existing rules.

- To load a filter ruleset into the DAG card use:



```
dagixp_filter_loader -d0 -x -f  ruleset.xml
```

- To display filtering statistics use:



```
dagixp_filter_loader -d0 -s -i3
```

- Start capturing the filtered data using:

```
dagsnap -d0 -v -o output_filename
```

For more information on the specific configuration options available for the IXP Filter please refer to the *IXP Filtering API User Guide* available from Endace Customer Support at support@endace.com.

Chapter 7: Synchronizing Clock Time

Overview

The Endace DAG cards have sophisticated time synchronisation capabilities, which allow for high quality timestamps, optionally synchronized to an external time standard.

The core of the DAG synchronisation capability is known as the DAG Universal Clock Kit (DUCK).

An independent clock in each DAG card runs from the PC clock. The card's clock is initialised using the PC clock, and then free-runs using a crystal oscillator.

Each card's clock can vary relative to a PC clock, or other DAG cards.

DUCK Configuration

The DUCK is designed to reduce time variance between sets of DAG cards or between DAG cards and coordinated universal time [UTC].

You can obtain an accurate time reference by connecting an external clock to the DAG card using the time synchronisation connector. Alternatively you can use the host PCs clock in software as a reference source without any additional hardware.

Each DAG card can also output a clock signal for use by other cards.

Common Synchronization

The DAG card time synchronisation connector supports a Pulse-Per-Second (PPS) input signal, using RS-422 signalling levels.

Common synchronisation sources include GPS or CDMA (cellular telephone) time receivers.

Endace also provides the TDS 2 Time Distribution Server modules and the TDS 6 units that enable you to connect multiple DAG cards to a single GPS or CDMA unit.

For more information please refer to the Endace website at http://www.endace.com/accessories.htm , or the *TDS 2/TDS 6 Units Installation Manual*.

Timestamps

ERF files contains a hardware generated timestamp of each packet's arrival. The arrival time can be either the point at which the start of the packet arrives (head) or the point at which the end of the packet arrives (tail).

See *Default Configuration* in *Chapter 3: Card Configuration* earlier in this user guide for more information on configuring the timestamp head/tail option

The format of this timestamp is a single little-endian 64-bit fixed point number, representing the number of seconds since midnight on the January 1970.

The high 32-bits contain the integer number of seconds, while the lower 32-bits contain the binary fraction of the second. This allows an ultimate resolution of 2-32 seconds, or approximately 233 picoseconds.

The ERF timestamp allows you to find the difference between two timestamps using a single 64-bit subtraction. You do not need to check for overflows between the two halves of the structure as you would need to do when comparing Unix time structures.

Different DAG cards have different actual resolutions. This is accommodated by the lowermost bits that are not active being set to zero. In this way the interpretation of the timestamp does not need to change when higher resolution clock hardware is available.

Example

Below is example code showing how a 64-bit ERF timestamp (erfts) can be converted into a `struct timeval` representation (tv):

```
unsigned long long lts;
struct timeval tv;

 lts = erfts;
 tv.tv_sec = lts >> 32;
 lts = ((lts & 0xffffffffULL) * 1000 * 1000);
 lts += (lts & 0x80000000ULL) << 1;      /* rounding */
 tv.tv_usec = lts >> 32;
 if(tv.tv_usec >= 1000000) {
     tv.tv_usec -= 1000000;
     tv.tv_sec += 1;
      }
```

Configuration Tools

The DUCK is very flexible, and can be used with or without an external time reference. It can accept synchronisation from several input sources, and also be made to drive its synchronisation output from one of several sources.

Synchronisation settings are controlled by the `dagclock` utility.

Note: You should only run `dagclock` after you have loaded the appropriate Xilinx images. If at any stage you reload the Xilinx images you must rerun `dagclock` to restore the configuration.

```
 dagclock -h
Usage: dagclock [-hvVxk] [-d dag] [-K <timeout>] [-l <threshold>] [option]
```

`-h`	`--help,--usage`	this page
`-v`	`--verbose`	increase verbosity
`-V`	`--version`	display version information
`-x`	`--clearstats`	clear clock statistics
`-k`	`--sync`	wait for duck to sync before exiting
`-d`	`dag`	the DAG device
`-K`	`timeout`	sync timeout in seconds, default 60
`-l`	`threshold`	health threshold in ns, default 596

Option:

`default`	RS422 in, none out
`none`	None in, none out
`rs422in`	RS422 input
`hostin`	Host input (unused)
`overin`	Internal input (synchronise to host clock)
`auxin`	Aux input (unused)
`rs422out`	Output the rs422 input signal
`loop`	Output the selected input
`hostout`	Output from host (unused)
`overout`	Internal output (master card)
`set`	Set DAG clock to PC clock
`reset`	Full clock reset. Load time from PC, set rs422in, none out

Note: By default, all DAG cards listen for synchronisation signals on their RS-422 port, and do not output any signal to that port

```
dagclock –d dag0
muxin   rs422
muxout  none
status  Synchronised Threshold 596ns Failures 0 Resyncs 0
error   Freq -30ppb Phase -60ns Worst Freq 75ppb Worst Phase 104ns
crystal Actual 100000028Hz Synthesized 67108864Hz
input   Total 3765 Bad 0 Singles Missed 5 Longest Sequence Missed 1
start   Thu Apr 28 13:32:45 2005
host    Thu Apr 28 14:35:35 2005
dag     Thu Apr 28 14:35:35 2005
```

Card with Reference

Overview

To obtain the best timestamp accuracy you should connect the DAG card to an external clock reference, such as a GPS or CDMA time receiver.

To use an external clock reference source, the host PC's clock must be accurate to UTC to within one second. This is used to initialise the DUCK.

When the external time reference source is connected to the DAG card time synchronisation connector, the card automatically synchronises to a valid signal.

Pulse Signal from External Source

The DAG time synchronisation connector supports an RS-422 (PPS) signal from an external source. This is derived directly from an external reference source, or distributed through the Endace TDS 2 (Time Distribution Server) module which allows two DAG cards to use a single receiver. It is also possible for more than two cards to use a single receiver by "daisy-chaining" TDS-6 expansion modules to the TDS-2 module. Each TDS-6 , module provides outputs for an additional 6 DAG cards.

Synchronise to an external source as follows:

```
dagclock –d dag0
muxin rs422
muxout none
status Synchronised Threshold 596ns Failures 0 Resyncs 0
error Freq 30ppb Phase -15ns Worst Freq 2092838ppb Worst Phase
33473626ns
crystal Actual 100000023Hz Synthesized 67108864Hz
input Total 225 Bad 0 Singles Missed 1 Longest Sequence Missed 1
start Thu Apr 28 14:55:20 2005
host Thu Apr 28 14:59:06 2005
dag Thu Apr 28 14:59:06 2005
```

Connecting the Time Distribution Server

You can connect the TDS 2 module to the DAG card using standard RJ-45 Ethernet cable including existing RJ-45 building cabling. The TDS may be located up to 600m (2000ft) from the DAG card depending upon the quality of the cable used, possible interference sources and other environmental factors. Please refer to the *TDS2/TDS6 User Guide* for more in formation

! **Caution:** Never connect a DAG card and/or the TDS 2 module to active Ethernet equipment or telephone equipment.

Testing the Signal

For Linux and FreeBSD, when a synchronisation source is connected the driver outputs messages to the console log file `/var/log/messages`.

To test the signal is being received correctly and has the correct polarity use the `dagpps` tool as follows:

```
dagpps –d dag0
```

`dagpps` measures the input state many times over several seconds, displaying the polarity and length of input pulse. The DAG 3.7T card also has an LED indicator for synchronisation (PPS) signals. See *Chapter 3: Configuring the Card* earlier in this User Guide for more information.

Single Card No Reference

When a single card is used with no external reference, the card can be synchronised to the host PC clock. Most PC clocks are not very accurate by themselves, but the DUCK drifts smoothly at the same rate as the PC clock.

If a PC is running NTP to synchronise its own clock, then the DUCK clock is not as smooth because the PC clock is adjusted in small jumps. However the DUCK clock does not drift away from UTC.

The synchronisation achieved with this method is not as accurate as using an external reference source such as GPS.

The DUCK clock is synchronized to a PC clock by setting input synchronization selector to overflow as follows:

```
dagclock –d dag0 none overin
muxin   overin
muxout  none
status  Synchronised Threshold 11921ns Failures 0 Resyncs 0
error   Freq 1836ppb Phase 605ns Worst Freq 143377ppb Worst
Phase 88424ns
crystal Actual 49999347Hz Synthesized 16777216Hz
input   Total 87039 Bad 0 Singles Missed 0 Longest Sequence
Missed 0
start   Wed Apr 27 14:27:41 2005
host    Thu Apr 28 14:38:20 2005
dag     Thu Apr 28 14:38:20 2005
```

Two Cards No Reference

Overview

If you are using two DAG cards in a single host PC with no reference clock, you must synchronise the cards using the same method if you wish to compare the timestamps between the two cards. You may wish to do this for example if the two cards monitor different directions of a single full-duplex link. You can synchronise the cards in two ways:

- One card can be a clock master for the second. This is useful if you want both cards to be accurately synchronised with each other, but not so for absolute time of packet time-stamps, or
- One card can synchronise to the host and also act as a master for the second card

Two Cards No Reference (cont.)

Synchronising with Each Other

Although the master card’s clock will drift against UTC, the cards will still be locked together. This is achieved by connecting the time synchronisation connectors of both cards using a standard RJ-45 Ethernet cross-over cable.

Configure one of the cards as the master so that the other defaults to being a slave as follows:

```
dagclock –d dag0 none overout
muxin   none
muxout  over
status  Not Synchronised Threshold 596ns Failures 0 Resyncs 0
error   Freq 0ppb Phase 0ns Worst Freq 0ppb Worst Phase 0ns
crystal Actual 100000000Hz Synthesized 67108864Hz
input   Total 0 Bad 0 Singles Missed 0 Longest Sequence Missed 0
start   Thu Apr 28 14:48:34 2005
host    Thu Apr 28 14:48:34 2005
dag     No active input - Free running
```

Note: The slave card configuration is not shown as the default configuration will work.

Synchronising with Host

To prevent the DAG card clock time-stamps drifting against UTC, the master can be synchronised to the host PC’s clock which in turn utilises NTP. This then provides a master signal to the slave card.

Configure one card to synchronize to the PC clock and output a RS-422 synchronization signal to the second card as follows:

```
dagclock –d dag0 none overin overout
muxin   over
muxout  over
status  Synchronised Threshold 11921ns Failures 0 Resyncs 0
error   Freq -691ppb Phase -394ns Worst Freq 143377ppb Worst Phase
88424ns
crystal Actual 49999354Hz Synthesized 16777216Hz
input   Total 87464 Bad 0 Singles Missed 0 Longest Sequence Missed 0
start   Wed Apr 27 14:27:41 2005
host    Thu Apr 28 14:59:14 2005
dag     Thu Apr 28 14:59:14 2005
```

The slave card configuration is not shown, the default configuration is sufficient.

Connector Pin-outs

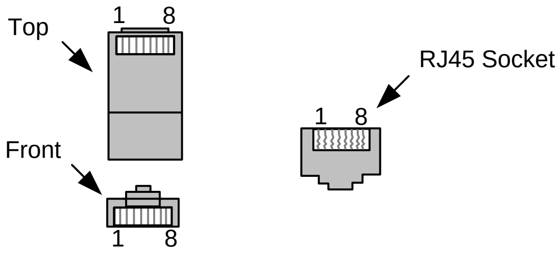
Overview

DAG cards have an 8-pin RJ45 connector with two bi-directional RS422 differential circuits, A and B. The PPS signal is carried on circuit A, and the serial packet is connected to the B circuit.

Pin Assignments

The 8-pin RJ45 connector pin assignments and plugs and sockets are shown below:

1. Out A+
2. Out A-
3. In A+
4. In B+
5. In B-
6. In A-
7. Out B+
8. Out B-



Normally you should connect the GPS input to the A channel input (pins 3 and 6).

The DAG card can also output a synchronization pulse for use when synchronizing two DAG cards without a GPS input. The synchronization pulse is output on the Out A channel (pins 1 and 2).

Ethernet Crossover Table

You can use a standard Ethernet crossover cable to connect the two cards as shown below:

TX_A+	1	3	RX_A+
TX_A-	2	6	RX-A-
RX_A+	3	1	TX_A+
RX_B+	4	7	TX_B+
RX_B-	5	8	TX_B-
RX_A-	6	2	TX_A-
TX_B+	7	4	RX_B+
TX_B-	8	5	RX_B-

Chapter 8: Data Formats

Overview

DAG Cards produce trace files in their own native format called ERF (Extensible Record Format). The ERF type depends upon the type of connection you are using to capture data.

The DAG 7.1S supports the following ERF Types:

ERF Type	Description
1	TYPE_HDLC_POS PoS with HDLC Frame Record
3	TYPE_ATM ATM Cell Record
4	TYPE_AAL5 Reassembled AAL5 Frame Record
5	TYPE_MC_HDLC: Multi-channel HDLC Frame Record
7	TYPE_MC_ATM Multi-Channel ATM Cell Record
9	TYPE_MC-AAL5 Multi Channel Reassembled AAL5 Frame Record
12	TYPE_MC-AAL2 Multi Channel Reassembled AAL2 Frame Record
18	TYPE_AAL2 Reassembled AAL2 Frame Record

The ERF file contains a series of ERF records with each record describing one packet. An ERF file consists only of ERF records, there is no special file header which allows concatenation and splitting to be performed arbitrarily on ERF record boundaries.

Generic Header

All ERF records share some common fields. Timestamps are in little-endian (Pentium native) byte order. All other fields are in big-endian (network) byte order. All payload data is captured as a byte stream, no byte or-ordering is applied.

The generic ERF header for concatenated links is shown below.

Byte 3	Byte 2	Byte 1	Byte 0
timestamp			
timestamp			
type	flags	rlen	
lctr/colour		wlen	
(rlen - 16) bytes of record			

Generic Header (cont.)

The generic ERF header for channelised links is shown below:

Byte 3	Byte 2	Byte 1	Byte 0
timestamp			
timestamp			
type	flags	rlen	
lctr/colour		wlen	
MCH (Multichannel header)			
(rlen - 20) bytes of packet			

timestamp The time of arrival of the cell, an ERF 64-bit timestamp. See *Timestamps* in *Chapter 5: Synchronising Clock Time* earlier in this User Guide for more information.

type One of the following:

1:	TYPE_HDLC_POS
3:	TYPE_ATM
4:	TYPE_AAL5
5:	TYPE_MC_HDLC
7:	TYPE_MC_ATM
9:	TYPE_MC_AAL5
12:	TYPE_MC_AAL2
18:	TYPE_AAL2

flags This byte is divided into several fields as follows:

1-0: Enumerates capture interface 0-3

2: Varying record lengths

3: Truncated record (insufficient buffer space)

4: RX error (link layer error)

5: DS error (internal error)

6: Reserved

7: General direction bit. This bit has two uses, it indicates from where a packet has arrived, either the host or line, and enables the XScale to target the packet at either the host or line. The direction bit can be interpreted in the context of either the Rx or Tx hole

In the XScale/Host Rx hole, a value of “1” indicates the ERF has arrived from the line. A value of “0” indicates the record was received from the host.

In the XScale Tx hole, a value of “1” tells the ERF Mux to direct packets to the line. Avalue of “0” directs packets to the host.

rlen Record length. Total length of the record transferred over the PCI bus to storage.

lctr Depending upon the ERF type this 16 bit field is either a loss counter of colour field. The loss counter records the number of packets lost between the DAG card and the memory hole due to overloading on the PCI bus.

wlen Wire length. Packet length including some protocol overhead. The exact interpretation of this quantity depends on physical medium.

Type-1 Record

The TYPE_HDLC_PoS variable length record for concatenated links is shown below:

Byte 3	Byte 2	Byte 1	Byte 0
timestamp			
timestamp			
type 1	flags	rlen	
lctr		wlen	
HDLC Header			
(rlen - 20) bytes of packet			

Type-3 Record

The TYPE_ATM record for concatenated links is shown below:

Byte 3	Byte 2	Byte 1	Byte 0
timestamp			
timestamp			
type 3	flags	rlen	
lctr		wlen	
ATM Header			
48 bytes of cell			

Note: This does not include the Header Error Correction (HEC) 8-bit checksum.

Type-4 Record

The format of the TYPE_AAL5_ record for concatenated links is shown below:

Byte 3	Byte 2	Byte 1	Byte 0
timestamp			
timestamp			
type 4	flags	rlen	
lctr		wlen	
ATM Header			
(rlen =20) bytes of AAL5 frame			

Note: This does not include the Header Error Correction (HEC) 8-bit checksum.

Type-5 Record

The Type 5 Multi-channel HDLC Frame record for channelised links is the same as the normal ERF Types but capture interface is always zero.

> **Note:** Fixed length mode is not supported. RX error is set if any MC header error bit is set.

The record is shown below:

<table>
<tr><th>BYTE 3</th><th>BYTE 2</th><th>BYTE 1</th><th>BYTE 0</th></tr>
<tr><td colspan="4">timestamp</td></tr>
<tr><td colspan="4">timestamp</td></tr>
<tr><td>type:5</td><td>flags</td><td colspan="2">rlen</td></tr>
<tr><td colspan="2">lctr</td><td colspan="2">wlen</td></tr>
<tr><td colspan="4">MC Header</td></tr>
<tr><td colspan="4">HDLC header</td></tr>
<tr><td colspan="4">(rlen – 24) bytes of packet</td></tr>
</table>

Note: Length may vary depending upon protocol

The Type 5 Multi-channel HDLC Frame record MC header is divided into several bit fields as shown below:

Bits	Attribute
0-9	Connection Number [0-511].
10-15	Reserved.
16-23	Reserved.
24	FCS Error.
25	Short Record Error [<5 Bytes].
26	Long Record Error [>2047 Bytes].
27	Aborted Frame Error.
28	Octet Error. The closing flag was not octet aligned after bit stuffing.
29	Lost Byte Error. The internal data path had an unrecoverable error.
30	1st Rec. This is the first record received since this connection was configured.
31	Reserved

Type-7 Record

The Type 7 Multi-channel ATM Cell record for channelised links is the same as the normal ERF Types but capture interface is always zero.

> **Note:** Fixed length mode is not supported. RX error is set if any MC header error bit is set.

The record is shown below:

<table>
<tr><th>BYTE 3</th><th>BYTE 2</th><th>BYTE 1</th><th>BYTE 0</th></tr>
<tr><td colspan="4">timestamp</td></tr>
<tr><td colspan="4">timestamp</td></tr>
<tr><td>type:7</td><td>flags</td><td colspan="2">rlen</td></tr>
<tr><td colspan="2">lctr</td><td colspan="2">wlen</td></tr>
<tr><td colspan="4">MC Header</td></tr>
<tr><td colspan="4">ATM header</td></tr>
<tr><td colspan="4">48 bytes of cell</td></tr>
</table>

Note: This does not include the Header Error Correction (HEC) 8-bit checksum

The Type 7 Multi-channel ATM MC header is divided into several bit fields as shown below:

Bits	Attribute
0-9	Connection number [0-511] or IMA group ID
10-14	Reserved.
15	Multiplexed from IMA into internal ATM. “`0`” for the DAG 7.1S
16-19	Physical port [0-15] cell was captured. `0x0` for the DAG 7.1S.
20-23	Reserved.
24	Lost Byte Error. The internal data path has an unrecoverable error.
25	HEC corrected.
26	OAM Cell CRC-10 Error [Not implemented].
27	OAM Cell
28	1st Rec. This is the first record received since this connection was configured.
29-31	Reserved.

> **Note:** When bit 15 of the MC Header is set, the bottom 9 bits (Connection Number/IMA ID) is treated as an IMA Group ID instead of a connection number.

Type 9 Record

The Type 9 Multi Channel Reassembled AAL5 Frame record for channelised links is the same as the normal ERF Types but capture interface is always zero.

> **Note:** Fixed length is not supported. RX error is set if any MC header error bit is set.

The record is shown below:

BYTE 3	BYTE 2	BYTE 1	BYTE 0
timestamp			
timestamp			
type9	flags	rlen	
lctr		wlen	
MC Header			
ATM Header			
(rlen - 24) bytes of AAL5 frame			

Note: This is the packet length of the ATM header and data, but does not include the padding required for 64 bit alignment.

Note: This does not include the Header Error Correction (HEC) 8-bit checksum

The Type 9 Multi Channel AAL5 Frame header is divided into several bit fields as shown below:

Bits	Attribute
0-9	Connection number (0-1023).
10-15	Reserved
16-19	Physical port (0-15) cell was captured on. `0x0` for the DAG 7.1S
20	CRC checked
21	CRC error
22	Length Checked
23	Length Error
24-27	Reserved
28	1st Cell. This is the first cell received since this connection was configured
29	Reserved
30	Reserved
31	Reserved

Type 12 Record

The Type 12 Multi Channel reassembled AAL2 Frame record for channelised links is the same as the normal ERF Types but capture interface is always zero.

> **Note:** Fixed length is not supported. RX error is set if any MC header error bit is set.

The record is shown below:

BYTE 3	BYTE 2	BYTE 1	BYTE 0
timestamp			
timestamp			
type12	flags	rlen	
lctr		wlen	
MC Header			
ATM Header			
(rlen - 24) bytes of AAL2 frame			

Note: This is the packet length of the ATM header and data, but does not include the padding required for 64 bit alignment.

The Type 12 Multi Channel AAL2 Frame header is divided into several bit fields as shown below:

Bits	Attribute
0-9	Connection number (0-1023).
10-12	Reserved for possible extra connection numbers
13-15	Reserved for indication of AAL2 type (a value of 0x0 indicates a SSSAR packet).
16-19	Physical port (0-15) cell was captured on. `0x0` for the DAG 7.1S
20	Reserved
21	1st Cell. This is the first cell received since this connection was configured.
22	MAAL Error (errnum as specified in ITU I.363.2 is copied to the data part of this record)
23	Length Error
24-31	Channel Identification Number (cid)

Type 18 Record

The Type 18 Reassembled AAL2 Frame record for concatenated links is shown below:

BYTE 3	BYTE 2	BYTE 1	BYTE 0
timestamp			
timestamp			
type 18	flags	rlen	
lctr		wlen	
AAL2 Extension Header			
ATM Header			
(rlen - 24) bytes of AAL2 frame			

The Type 18 ATM Extension header is divided into several bit fields as shown below:

Bits	Attribute
0-7	Channel identification number
8-15	MAAL error (errnum as specified in ITU I.363.2 is copied to the data part of this record)
16-23	AAL2 flags
24-31	Reserved

The Type 18 Flags byte is divided in to several bit fields as shown below:

Bits	Attribute
0	MAAL Error Indication, will be set if the frame has a MAAL error otherwise it is cleared.
1-7	Reserved

Chapter 9
Troubleshooting

Reporting Problems

If you have problems with a DAG card or Endace supplied software which you are unable to resolve, please contact Endace Customer Support at support@endace.com.

Supplying as much information as possible enables Endace Customer Support to be more effective in their response to you. The exact information available to you for troubleshooting and analysis may be limited by nature of the problem. However the following items will assist a quick resolution:

- DAG card[s] model and serial number.
- Host PC type and configuration.
- Host PC operating system version
- DAG software version package in use
- Any compiler errors or warnings when building DAG driver or tools
- For Linux and FreeBSD, messages generated when DAG device driver is loaded. These can be collected from command `dmesg`, or from log file `/var/log/syslog`.
- Output of daginf
- Firmware versions from `dagrom –x`.
- Physical layer status reported by: `dagconfig`
- Network link statistics reported by: `dagconfig –si`
- Network link configuration from the router where available.
- Contents of any scripts in use.
- Complete output of session where error occurred including any error messages from DAG tools. The `typescript` Unix utility may be useful for recording this information.
- A small section of captured packet trace illustrating the problem.